

HEIDENHAIN

Programming Manual



Programming Manual Milling V310-V330

MillPlus

HEIDENHAIN NUMERIC B.V.
Eindhoven (NL)
Tel: 31.40.278 49 82
Fax: 31.40.278 59 34

12/97

© HEIDENHAIN NUMERIC B.V. EINDHOVEN, THE NETHERLANDS 1997

The publisher accepts no liability in respect of specifications on the basis of the information contained in these instructions.

For the specifications of the numerical controller, please refer to the order data and corresponding specification description only.

All rights reserved. Copying of this manual or parts thereof only permitted with the written consent of the copyright holder.

1. Introduction	1
1.1 Foreword	1
1.2 Companion manuals	1
2. General programming information	3
2.1 About part programs	3
2.1.1 Program words	4
2.1.2 Program blocks	5
2.1.3 Writing a part programm	6
2.1.4 Zero points	7
2.2 Axes Configurations on machine tools	10
2.2.1 Defining Coordinates	12
2.2.2 Cartesian Coordinates	12
2.2.3 Polar Coordinates	13
2.2.4 Combining a Linear Coordinate and Angle	14
3. About this manual	15
3.1 Philosophy and purpose of the manual	15
3.2 Contents of each section	15
3.2.1 G-functions	15
3.2.2 LAF	15
3.2.3 F-functions	15
3.2.4 H-functions	16
3.2.5 M-functions	16
3.2.6 S-function	16
3.2.7 T-function	16
3.2.8 E-parameter	16
3.2.9 Geometric calculations with continuous movements	16
3.2.10 Graphical support	16
3.2.11 Tool memory	16
3.2.12 Machine constants	16
3.2.13 External program selection	16
3.2.14 The V320 improvements over it's predecessors	16
3.2.15 Application notes appendix.	16
3.3 Programming functions.	17
4. Rapid traverse G0	19
5. Linear interpolation G1	23
6. Circular interpolation (CW/CCW) G2/G3	35
7. Dwell time G4	53
8. Spline interpolation G6	55
9. Defining the pole point (size reference point) G9 (from V320)	63
10. One/Two point or two line geometry with chamfer or rounding G11	69
11. Repeat function G14	83
12. Mainplane XY, tool Z G17	87
13. Main plane XZ, tool Y G18	91
14. Mainplane YZ, tool X G19	95

Table of Contents

15. Macro call G22	99
16. Program call G23	105
17. Enable/Disable feed override G25/G26	107
18. Positioning functions G27/G28 active till V320	109
19. Positioning functions G27/G28 active from V320	115
20. Conditional jump G29	119
21. Activate/ deactivate compensation (active in V320) G39	123
22. Cancel tool radius compensation G40	127
23. Tool radius compensation (left/right) G41/G42	131
24. Tool radius comp. to/past endpoint G43/G44	141
25. Axis parallel measuring movement G45	145
26. Measure tool dimensions G45+ M25	151
27. Measuring a full circle G46	153
28. Probe calibration G46+ M26	159
29. Checking on tolerances G49	161
30. Processing measuring results G50	167
31. Cancel/Activate G52 zero point shift G51/G52 (till V310)	175
32. Cancel/Activate pallet zero point shift G51/G52 (from V310)	177
33. Cancel/Activate zero point shift G53/G54 - G59 (MC84=0)	179
34. Zero point shift extension G54 MC84>0 (from V320)	183
35. Tangential approach G61	187
36. Tangential exit G62	193
37. Cancel/Activate geometric calculations G63/G64	199
38. Select negative/positive tool direction G66/G67	221
39. Inch/Metric programming G70/G71	225
40. Cancel/Activate scaling or mirror imaging G72/G73	227
41. Programmable absolute position G74	235
42. Bolt hole circle G77	239
43. Point definition G78	245

44. Activate cycle G79	249
45. Drilling cycle G81	253
46. Deep hole drilling cycle G83	257
47. Tapping cycle G84	263
48. Reaming cycle G85	267
49. Boring cycle G86	271
50. Rectangular pocket milling cycle G87	275
51. Groove milling cycle G88	281
52. Circular pocket milling cycle G89	287
53. Absolute/Incremental programming G90/G91	293
54. Wordwise absolute and incremental programming (from V320)	297
55. Incremental/Absolute zero point shift G92/G93	299
56. Select feedrate unit G94/G95	307
57. Graphic window definition G98	309
58. Definition of workpiece blank as a box G99	311
59. 3D-Tool correction G141	313
60. Linear measuring movement G145	321
61. Read probe status G148	341
62. Read tool data and zero offset G149	343
63. Write tool data and zero offset G150	347
64. Basic coordinate system G180	351
65. Cylindrical coordinate system G182	353
66. Graphic window definition G195	363
67. End contour description G196	365
68. Begin inner/outer contour description G197/G198	367
69. Begin contour description G199	375
70. Create pocket cycle macro's G200	383
70.1 Introduction universal pocket cycle	383
70.2 Partprogram structure	385
70.3 Translation, rotation and mirror image of a pocket	386
70.4 Same pocket in another program	387
70.5 Operating section	388

Table of Contents

70.6 Error messages	391
70.7 Example	393
71. Start contour pocket cycle G201	397
71.1 Usage of the generated macros	399
71.1.1 Starting points macro	399
71.1.2 Machining macro	400
71.1.3 Macro for finishing a pocket contour	400
71.1.3.1 Feed-in point	400
71.1.3.2 General format of the macro for a circular feed-in	401
71.1.3.3 Circular pocket contour	402
71.1.4 Linear feed-in movements	403
71.1.5 Sequence of the macros on the machine	403
72. End contour pocket cycle G202	409
73. Start pocket contour description G203	413
74. End pocket contour description G204	417
75. Start island contour description G205	419
76. End pocket contour description G206	425
77. Call island contour macro G207	429
78. Quadrangle contour description G208	431
79. Programming error messages G300	439
80. Error in a program or macro G301	441
81. Calling machine constant values G322	443
82. Calling actual axes-positions values G326	445
83. Look Ahead Feed (LAF) function (starting from V320)	447
83.1 Introduction	447
83.2 Detailed specification	447
83.3 General theoretical description	449
84. F-Functions	453
85. Auxiliary function H	457
86. Program stop M0	459
87. Optional program stop M1	461
88. Spindle rotating clockwise/counter clockwise M3/M4	463
89. Spindle stop M5	465
90. Automatic tool change M6	467
91. Switch on coolant supplynr. 2 / Nr. 1. M7/M8	469
92. Switch on coolant supplynr. Nr. 1. M8	471

93. Switch off coolant supply M9	473
94. Switch on nr. 1 Coolant and rotate spindle CW/CCW M13/M14	475
95. Oriented spindle stop M19	477
96. Measuring tool sizes M25	479
97. Calibrating the measuring probe M26	481
98. Switch on / off a measuring probe M27/M28	483
99. End of partprogram M30	485
100. Select spindle speed range M41/M42/M43/M44	487
101. Manual tool change M66	489
102. Change tool compensation values M67	491
103. Spindle speed S	493
104. Tool number T	495
105. E-parameters	499
106. Geometric calculations with continuous movements	511
106.1 Geometric calculations with non-continuous movements G64	554
107. Graphical support	587
108. Tool memory	591
109. Machine constants	595
110. External program selection	599
111. The V330 improvements over it's predecessors	601
112. Application notes	603
112.1 Reaming cycle with separate outfeed	603
112.2 Boring cycle without dragline	607
112.3 Back boring cycle	611
112.4 Linear pattern	615
112.5 Circular pattern	620
112.6 Grid pattern	625
112.7 Staggered grid pattern	630
112.8 Finishing a circular pocket or island	635
112.9 Milling a circular groove	643
112.10 Rhomboidal pattern	647
112.11 Roughing a circular pocket	652
112.12 A circular pocket with spherical bottom	659
113. Index	667

1. Introduction

1.1 Foreword

This manual assist you in programming the controller.

The machine should not be operated, even for a short period, by anyone who has not received the **necessary training** either in the Company, at an Institute of Further Education or in one of the Training Centres.

Please follow this advice to ensure proper usage.

The controller and the machine are coordinated using machine constants. Some of these constants are accessible to the user. **Caution!**

A thorough understanding of the significance and functions of these constants is required if they are to be changed. If in doubt please contact our Customer Service Department.

Users should therefore always output their programs and specific data (e.g. technical data, machine constants etc.) on their PC or onto diskette. This prevents data from being lost irretrievably if the battery or back-up battery is defective.

We reserve the right to change the design, equipment and accessories in the interest of further development. No liability will be accepted for any errors in the data, illustrations or descriptions.

1.2 Companion manuals

The information relating to the installation, interfacing, operation, and programming for the controller cannot be adequately described in a single manual. Therefore, several manuals have been designed to give the user information relating to a particular type of task. The set of manuals available for the controller is listed in this section.

- User Manual
- CDS Manual (CNC Data Station Manual)
- Installation Manuals
- Interfacing
 - MIPS (Machine Interface Programming System.)
 - Basic IPLC Program

2. General programming information

2.1 About part programs

A part program is the complete set of data and instructions required for producing a particular work-piece on a numerically controlled machine tool.

The instructions may contain different operations, such as milling, drilling, tapping, etc. Each separate operation is a unit which can be split up into smaller instructions. One program block specifies one complete operation. The words in a block define the smaller instructions.

The proper machining sequence, with all the separate instructions, must be stated in a part program. Examples of separate instructions are tool movements, machine tool functions and technological data.

A program cannot be executed until it has been properly stored in the CNC system memory. A part program can be created and stored into the CNC memory in different ways:

1. Use interactive contour programming (ICP) for complex contours.
2. Use interactive part programming (IPP) for programming without knowledge of DIN programming.
3. Enter the program manually via the control panel.
4. Create the program separate from the control, use data terminal equipment to produce a data carrier (such as a punched paper tape, a magnetic digital cassette or disk) and input the data into the CNC memory.

2.1.1 Program words

The CNC PILOT control system employs the standard WORD ADDRESS system in which a word has two parts:

1. the address, which can be a single address (one alpha character) or an indexed address. An indexed address has an alpha character followed by an index and the character =, e.g. E1=.
2. a multi-digit number.

Words do not need leading zeros. However, if the value of a word is zero, then at least one zero must be written.

The words stating dimensional information can have a plus or a minus sign. If no sign is programmed, a positive value is assumed. A negative value must have a minus sign. Dimension words can be written with a decimal point; therefore trailing zeros need not be stated. The control system assumes that the decimal point is behind the last digit of the number if the decimal point is not stated.

NON-MODAL WORDS

Example of a single and an indexed word

Single word: X-21.43

'X' is the address, '-' is the sign and '21.43' the decimal number.

Indexed word: X1=-21.43 .

'X1=' is the address, '-' is the sign and '21.43' the decimal number.

2.1.2 Program blocks

A block can include several words considered as a unit which contains all the information needed for one complete operation or function. This operation can be a tool movement or a machine tool function, or a combination of both.

The CNC PILOT control system employs a VARIABLE BLOCK FORMAT. The block lengths can be different because of changes in the number or length of the words. A block can contain up to 255 characters.

The N-word must always be the first word in a block. The other words can be written in any order. The example gives the preferred sequence for the frequently used words.

Each word can occur only once in a block. Words such as E1= and E2= have different addresses and therefore can be both present in the same block.

On a data carrier the character line feed [LF] separates the blocks.

Example of a program block

```
N20 G1 X14 Z62.5 F300 S200 T12 M3
```

N20: Block number

G1: Preparatory function

X14 and Z62,5: Dimensional information

Technological and machine data such as spindle speed (S), feed rate (F), tool selection (T) and e.g. a direction of spindle rotation (M3) may be included as well.

THE BLOCK NUMBER N

The first word in a block is the block number which identifies that block. Each block must have a separate number. The block numbers range from N0 to N9999999.

A general rule is that a block number cannot be in the same program more than once. However, the check on the block numbers is inactive if a machine constant is set or the BTR possibility is used. The machine constant setting is useful when large programs should be executed and the BTR possibility is not used.

Block numbers can be in any sequence. The execution will be in the programmed sequence e.g.

programmed sequence: N10, N50, N30

executed sequence: N10, N50, N30

The **re-number** function of the control allows the block numbers to be automatically renumbered in increasing order, starting from N1.

The CNC system automatically generates block numbers when the programmer uses the control panel to input programs.

2.1.3 Writing a part programm

PROGRAM IDENTIFICATION

Each part program or subprogram has to start with an identification number which ranges from 1 to 9999999.

So numbers as 1, 125, 9001, 12345, 876543, 3451592 are valid identification numbers.

The rename function of the control is available for changing the identification number.

A part program name can be written between the characters CONTROL OUT '(' and CONTROL IN ')' and immediately after the identification number. These names are listed if the file directory is displayed on the control.

Example of part program identification with name

N9001 (PLATE NR. A334)

The following part program identification is possible for earlier CNC systems (compatibility) (for programs %PM... and macros %MM...)

%PM9001

N9001 (PLATE NR. A334)

These programs are automatically identified and stored correctly by the CNC system. Data transmission from the CNC to the outside is controlled by machine constants (MC795, MC799)

PART PROGRAM SETUP

To write a part program the programmer must do the following:

1. Determine the mounting of the workpiece and the position of the clamps
2. Determine the machining operation sequence
3. Determine the tools required for the operations
4. Determine for each tool the appropriate technological data
5. Determine the workpiece dimensions and the necessary movements.

The points 1 to 4 are outside the scope of this manual.

The movements on the machine are a combination of tool and workpiece movements. To simplify the programming the programmer should assume that all movements are tool movements. The configuration of the machine tool and CNC system determines how the movements are actually performed.

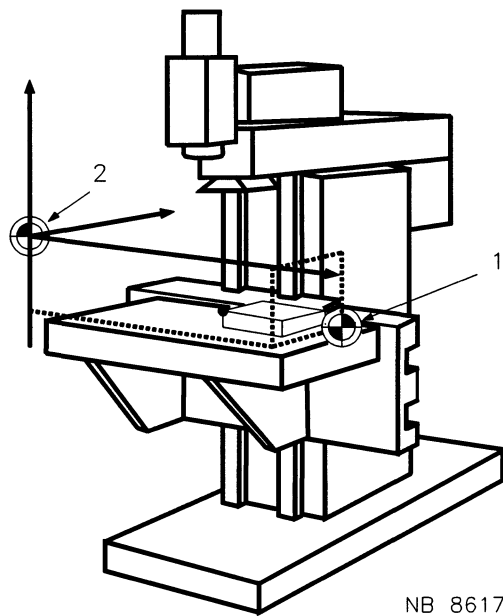
An imaginary coordinate system is positioned on the workpiece so that programmed movements refer to a zero. The programmer determines the position of this point in such a manner that the easiest programming calculations are produced. Refer to Axes configurations on machine tools for the directions of the coordinate axes.

PROGRAM STORAGE

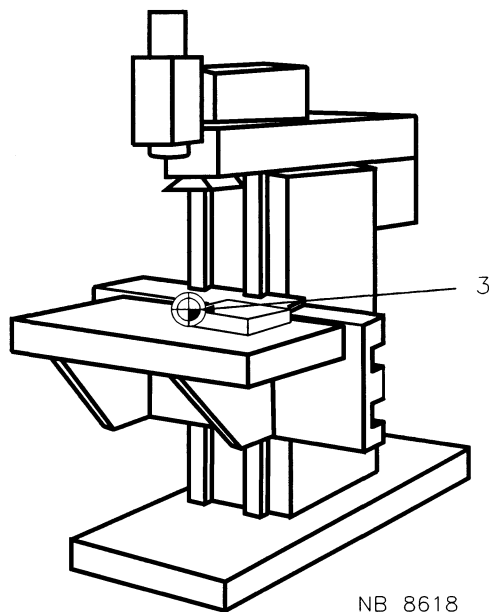
The user memory can store more than one part program or subprogram (macro). The actual number of stored programs and macros depends upon the size of each program or macro and on the available memory capacity. A machine constant sets the maximum amount of programs to be between 16 and 1000.

With the lock function it is possible to protect part programs and macros against unauthorised editing on the control.

2.1.4 Zero points



NB 8617

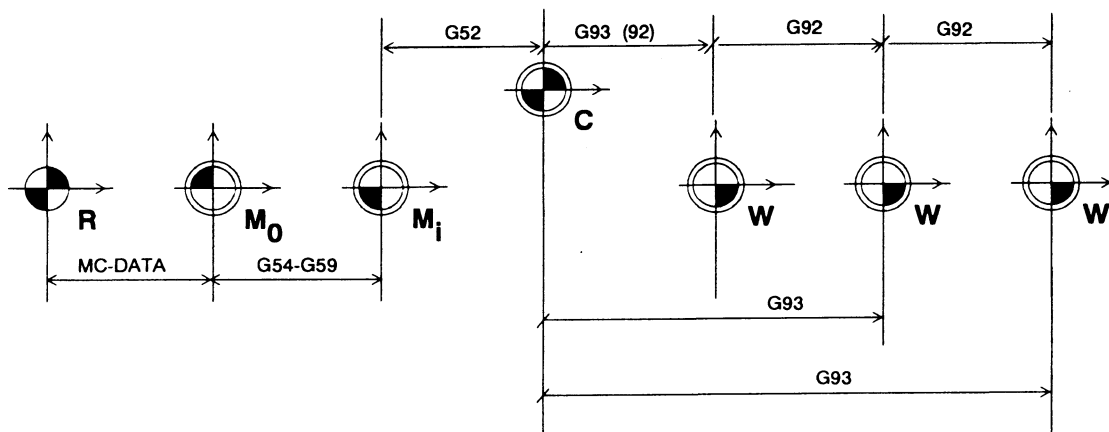


NB 8618

- 1= Machine reference point (R)
- 2= Machine zero point (M_0)
- 3= Program zero point (W)

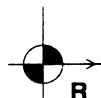
After power-on, REFERENCE POINT SEARCH must be carried out first. As a result, the machine zero point is known, since the zero offsets from the machine zero point (M_0) to the machine reference point (R) are stored as machine constants.

The part programmer establishes a program zero point (W) which is related to the part and from which the part dimensions are measured. This program zero point must also be related to the machine zero point which can be established with the functions G52 and/or G54-G59.



1. Machine reference point (R)

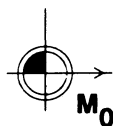
Each axis of a machine tool has a fixed point called the reference point of the axis. The reference points of all axes form the machine reference point (R).



During REFERENCE POINT SEARCH (refer to the Operating Manual) the tool moves to the reference point of the selected axis (or axes). When the reference point is reached, the axis is automatically zeroed by the control and the positions of the software limit switches are set.

2. Machine zero point (M_0)

The machine zero point is also a fixed point on the machine.



When the CNC control system is commissioned, the distances from the machine reference point (R) to the machine zero point (M_0) are measured along the axes and stored in the machine constant memory. Each axis has its own machine constant for this purpose.

After the machine reference point is established by REFERENCE POINT SEARCH, the control system reads the associated dimensions from the machine constant memory. The machine zero point (M_0) is set as the origin of the coordinate system and the displayed positions are related to this zero point.

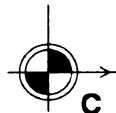
3. Secondary machine zero point (M_i)



When the machine tool has several clamping stations (e.g. pallet stations) each clamping station must have its own fixed zero point. These fixed zero points are called secondary machine zero points (M_i).

The zero offset memory contains the axis distances between the machine zero point (M_0) to the secondary machine zero points (M_i). Six secondary points can be stored by using the G54 to G59, G54I[0..99](from V320) functions.

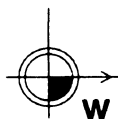
4. Mounting zero point (C)



When a secondary machine zero point (M_i) is established, the zero point of the mounting device must be determined. This zero point may coincide with the active M_i or can be set by the G52 PRESET AXIS function.

Zero point C is automatically set by the control, when an external program call with offset values is made.

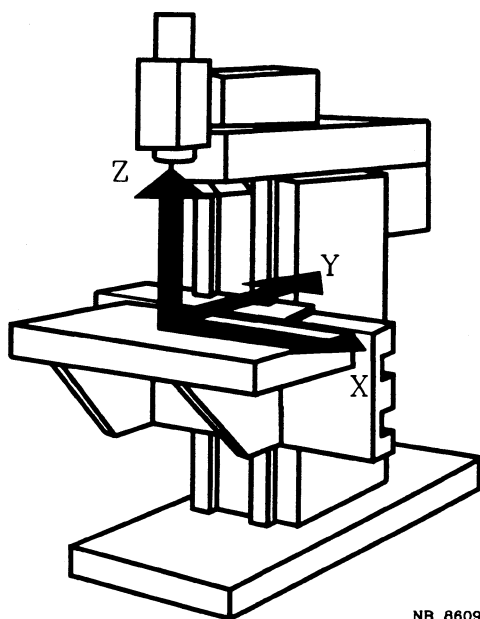
5. Program zero point (W)



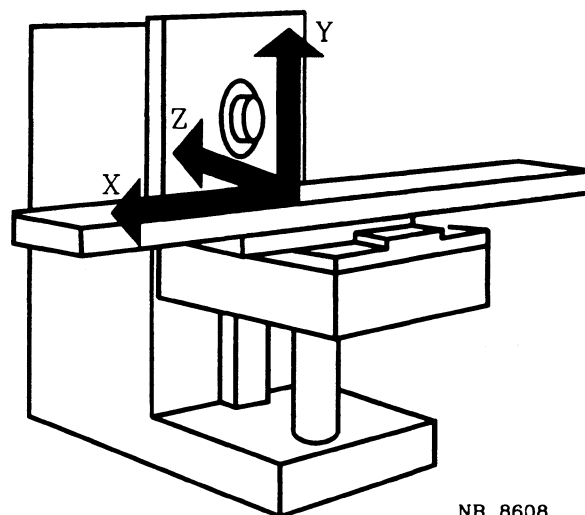
The program zero point W is the zero point from which the axis coordinates in the part program are measured. The position of point W can be set arbitrarily by the programmer. The position should be chosen in order that additional calculations for programming the workpiece be kept to the minimum.

The functions G52, G54-G59, G54I[0..99](from V320) and G92/G93 establish the relation between the program zero points and the machine zero point.

2.2 Axes Configurations on machine tools

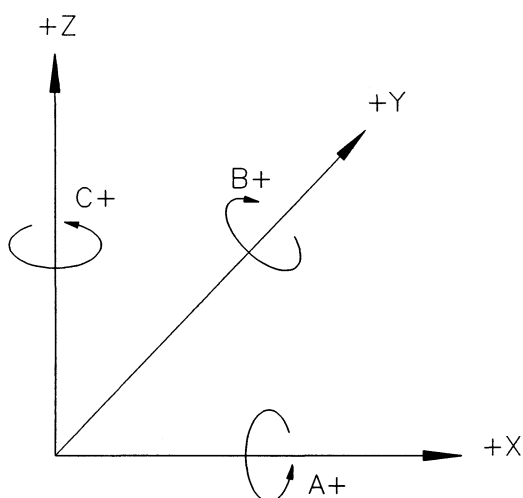


Vertical knee milling machine

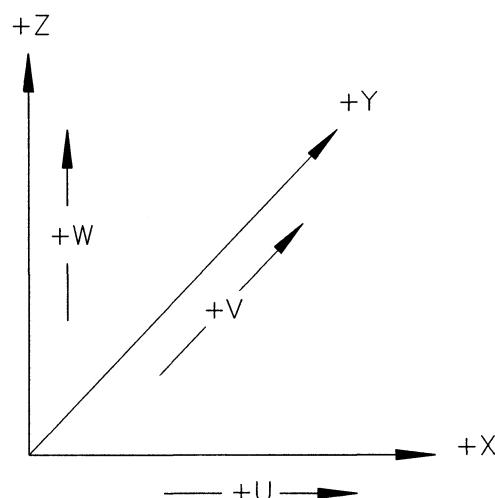


Horizontal knee milling machine

A milling machine has three main linear axes (X,Y,Z) which are at 90° to each other. The orientation of these axes is established by the Z-axis which is always parallel to the main spindle of the machine tool. The X-axis is horizontal and parallel to the work holding surface. Each main axis can have a rotary axis and a linear axis, parallel to a main axis. These are shown in the illustration below.



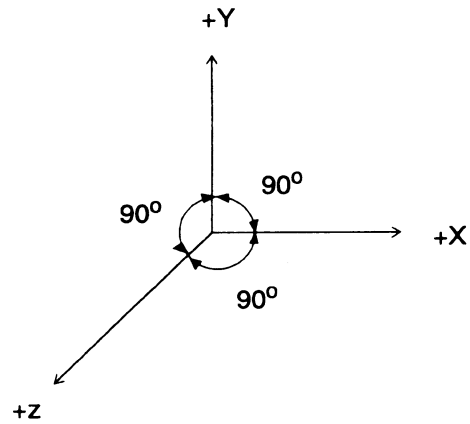
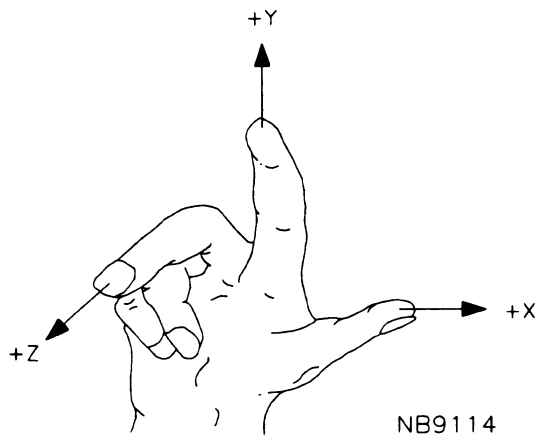
Rotary axes



Linear axes parallel to main axes

Orientation of main axes, rotary axes and linear parallel axes.

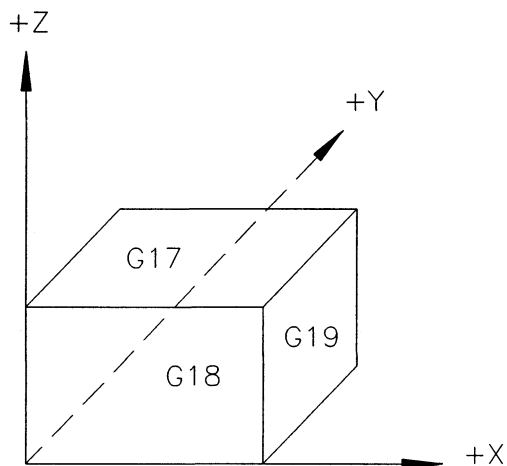
The standards ISO 841, DIN 66217 and EIA RS-267-A, all define the positions of axes on a numerically controlled machine. The right-hand rule is used for stating the orientation for all CNC machines axes.



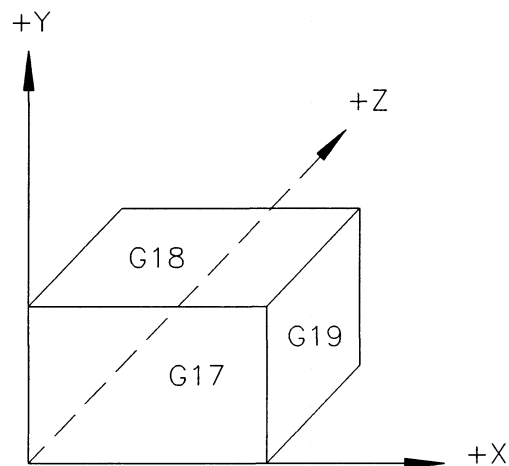
The thumb represents the X-axis, the forefinger the Y-axis and the middle finger the Z-axis. The directions in which the fingers are pointing represent the positive directions along the axes.

2.2.1 Defining Coordinates

Axes distances define the coordinates of points in three-dimensional(3-D) space. Axis coordinates will be in one of three planes (XY-plane, XZ-plane, YZ-plane).



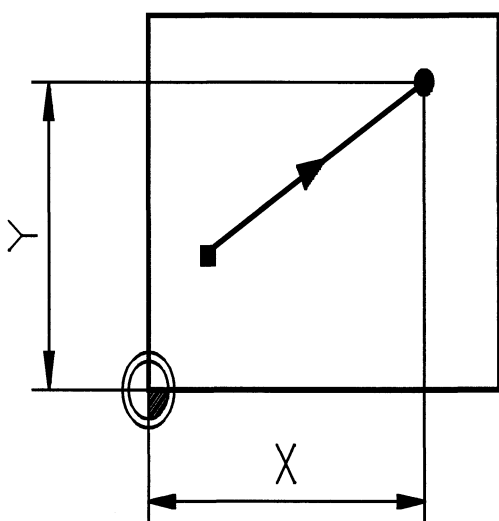
Vertical milling machine's planes



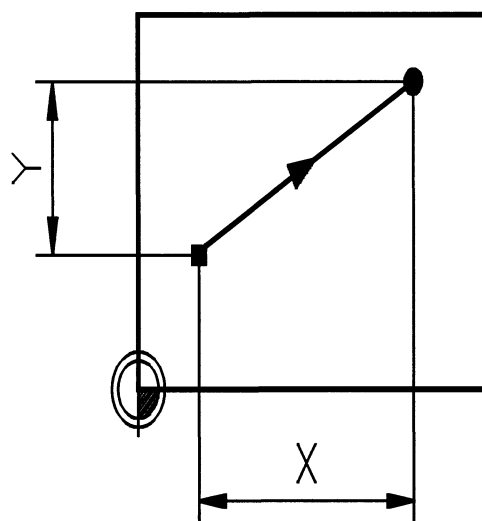
Horizontal milling machine's planes

NB8571

2.2.2 Cartesian Coordinates

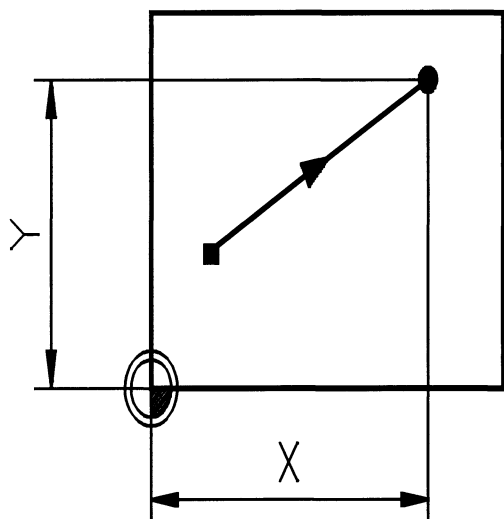


Absolute (G90) coordinates

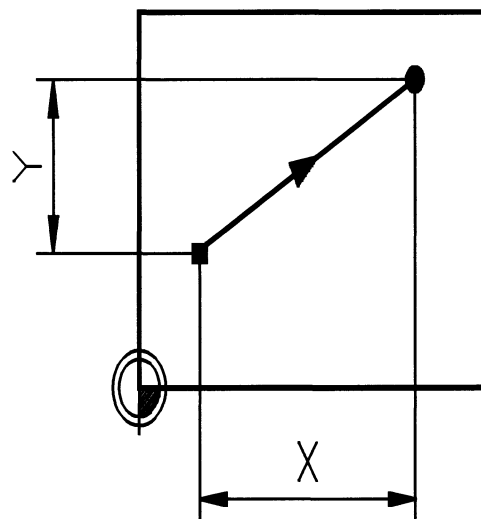


Incremental (G91) coordinates

2.2.3 Polar Coordinates

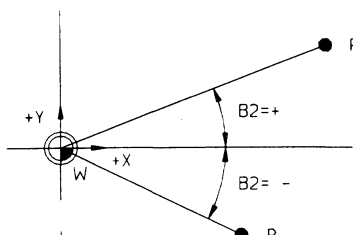


Absolute (G90) coordinates

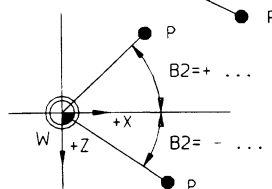


Incremental (G91) coordinates

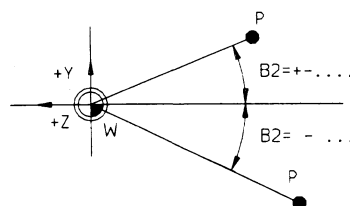
XY-PLANE (G17)



XZ-PLANE (G18)



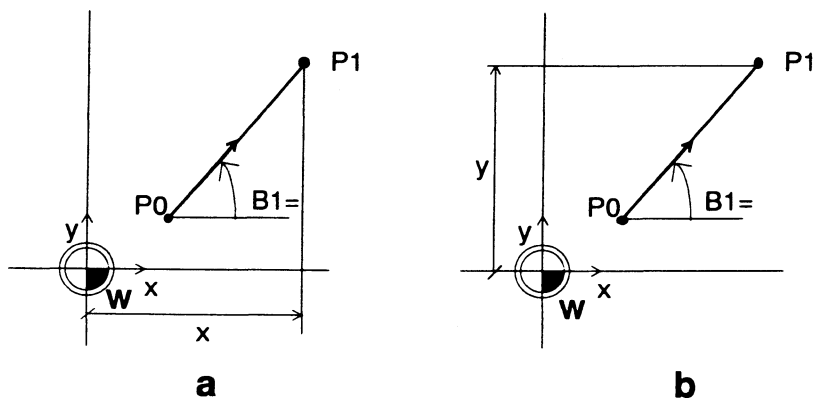
YZ-PLANE (G19)



NB6816/7/8

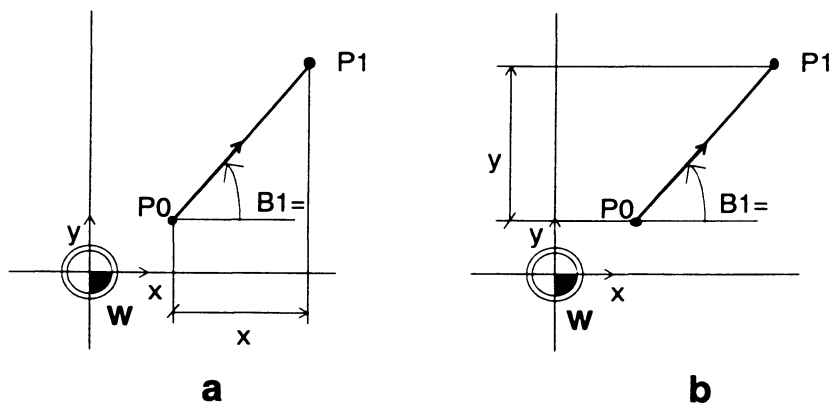
2.2.4 Combining a Linear Coordinate and Angle

One linear axis coordinate and an angle can, in combination, define a point's position.



NB7958

Absolute (G90) coordinates



NB6829

Incremental (G91) coordinates

3. About this manual

3.1 Philosophy and purpose of the manual

This manual has been arranged to allow access to comprehensive information relating to programs for the CNC PILOT system.

The core of the reference data in this manual is contained within the sections which describe the F, G, H, M, S and T- functions. Additional information, such as mathematical operations and formula, is contained in appendices.

3.2 Contents of each section

3.2.1 G-functions

G-functions are functions preparing the CNC controlled machine to programming instructions and are therefore named Preparatory functions. The contents of each section describing a G- function is placed under standard headings, which are:

Purpose

The reason(s) for using the function.

Format

The format(s) of a program block containing the function. The conventions used under this heading are given in the Introduction section of this manual.

Parameters

Program words defining the extent of the function's influence, or words which can be programmed when the function is active.

Associated functions

Functions which are in the same functional group, they can cancel the function.

Type of functions

Explains if the function is modal or not.

Notes and usage

Explanation(s) of how the function is used and under what circumstances.

Examples

Examples showing practical applications of the function.

3.2.2 LAF

Explains how LAF (Look Ahead Feed) has to be operated and how it functions.

3.2.3 F-functions

F-functions are functions which establish the feedrate (mm/min or inch/min). The feed rate is dependent of the situation.

3.2.4 H-functions

H-functions are assigned tasks by the machine tool builder. The programmer must therefore read the machine tool builder's documentation for description(s) of how this auxiliary function is used.

3.2.5 M-functions

M-functions are functions which directly affect CNC machine operations, e.g. switching the coolant supply on and off.

3.2.6 S-function

The S-function specifies the spindle speed in revolutions per minute (RPM).

3.2.7 T-function

The T-function specifies the number which is used to select a tool and also to store its dimensions in the CNC's Tool Memory.

3.2.8 E-parameter

E-parameter are useful in making a program more flexible. One program can be used for different products.

3.2.9 Geometric calculations with continuous movements

Geometric calculations with continuous movements is a function which enables the operator to make a program without knowing the exact coordinates of a certain point.

3.2.10 Graphical support

Graphical support visualizes the movement of the tool in different ways on the display.

3.2.11 Tool memory

This chapter explains how tool dimensions and other tool related parameters can be stored in the tool memory of the controller.

3.2.12 Machine constants

This chapter explains how the machine manufacturer can customize the controller for his machines.

3.2.13 External program selection

This chapter explains the use of automatic workpiece changers.

3.2.14 The V320 improvements over its predecessors

This chapter explains the difference between the V320 and its predecessors.

3.2.15 Application notes appendix.

This appendix contains a.o. explanations on the use of E- parameters and additional arithmetical operations which are available in the CNC software.

3.3 Programming functions.

Fundamentals of CNC Programming

Coordinate Measurement Modes

- G90/G91 - Absolute/incremental programming.
- G70/G71 - Inch/metric programming.

Basic Tool Movements

- G0 - Rapid traverse.
- G1 - Linear interpolation.
- Tool movement with a feedrate (in a linear and a rotary movement)
- Tool movement with a feedrate (3D-interpolation)
- G2/G3 - Circular interpolation (CW/CCW).
- G78 - Point definition.

Radius Compensation

- G41/G42 - Tool Radius Compensation (Left/Right).
- G43/G44 - Tool Radius Comp. TO/PAST End Point.
- G40 - Cancel Tool Radius Compensation.

Main Planes

- G17 - Main plane XY, tool Z.
- G18 - Main plane XZ, tool Y.
- G19 - Main plane YZ, tool X.

Positioning & Feedrate Functions

- G27/G28 - Cancel/activate positioning function.
- G25/G26 - Feed override active/inactive.
- G94/G95 - Select feed rate unit.
- G4 - Dwell time.

Tool & Spindle Speed Functions

- S - Spindle speed.
- T - Tool number.

Zero Datum Points

- G51/G52 - Cancel/activate G52 zero point shift.
- G53/G54-G59 - Cancel/activate zero point shift
- G54 I[0..99] - Activate zero point shift
- G92/G93 - Incremental/absolute zero point shift.

Graphical Simulations

- G195 - Graphic window definition.
- G196 - End contour description.
- G197 - Begin inner contour description.
- G198 - Begin outer contour description.
- G199 - Begin contour description.

Machine Functions

- M3/M4 - Spindle clockwise/counter-clockwise.
- M5 - Spindle stop.
- M19 - Orientated spindle stop.
- M7 - Switch on number 2 coolant supply.
- M8 - Switch on number 1 coolant supply.
- M9 - Switch off coolant supply.
- M13 - Switch on No. 1 coolant rotate spindle clockwise.
- M14 - Switch on No. 1 coolant rotate spindle counter-clockwise.
- M6 - Automatic tool change.

- M0 - Program stop.
- M1 - Optional program stop.
- M30 - Part program end.

Geometric Functions

- G11 - Polar coordinate, corner rounding, chamfer.
- G63/G64 - Cancel/activate geometric calculations.
- G72/G73 - Cancel/activate scaling or mirror imaging.

Defined (Canned) Cycles

- G79 - Activate cycle.
- G77 - Bolt hole circle.
- G81 - Drilling cycle.
- G83 - Deep hole drilling cycle.
- G84 - Tapping cycle.
- G85 - Reaming cycle.
- G86 - Boring cycle.
- G87 - Rectangular pocket milling cycle.
- G88 - Slot milling cycle.
- G89 - Circular pocket milling cycle.

Transfer of Program Control

- G14 - Repeat function.
- G29 - Conditional jump.
- G22 - Macro call.
- G23 - Program call.

E-parameters and optional arithmetical operations

(Refer to Appendices).

- G322 - Change a C(E-parameter) value in a machine constant.
(active in V320)

Special Functions

- G6 - Spline interpolation.
- G9 - Polar point definition (active in V320)
- G39 - Activate/deactivate compensation (active in V320)
- G141 - 3D tool correction.
- G180/G182 - Cancel/activate cylinder interpolation.

Measuring Cycles

- G45 M25 - Measure tool dimensions.
- G46 M26 - Probe calibration.
- G49 - Checking on tolerances.
- G50 - Processing measuring results.
- G145 - Linear measuring movement.
- G148 - Read probe status.
- G149 - Read tool data and offsets.
- G150 - Write tool data and offsets.

Auxiliary Function

- H - Auxiliary function.

4. Rapid traverse G0

Purpose

To use the rapid traverse rate for axis movements. This traverse rate is set by Machine Constants (per axis). The function G0 is used mainly for positioning a tool before and after cutting passes.

Format

N... G0 [axis coordinates]

Parameter

End point coordinates

X,Y,Z Endpoint coordinate

A,B,C Endpoint angle

P Point definition number (P1-P4)

P1= Point definition number

B1= Angle

L1= Path length

B2= Polar angle

L2= Polar length

A40= Radius A-axis

B40= Radius B-axis

C40= Radius rotary-axis

D Angle oriented spindle stop

For absolute and incremental programming

X90=,Y90=,Z90= Absolute endpoint

A90=,B90=,C90= Absolute endpoint angle

X91=,Y91=,Z91= Incremental endpoint

A91=,B91=,C91= Incremental endpoint angle

Modal words

F,S,T,M,H,E..=

F1=, T1=, T2=

Associated functions

G1, G2/G3, G6, G74.

Wordwise absolute/incremental programming (X90=..., X91=...)

Type of function

Modal

Notes and usage**DEFAULT MODE**

The G0 function is automatically set at the start of a program or after CLEAR CONTROL.

CANCELLATION

The function G0 is cancelled by a G1, G2, G3, G6.

STOP AFTER A RAPID MOVEMENT

The programmed position is reached before the next movement starts. So a stop occurs after a rapid movement.

NO STOP AFTER A RAPID MOVEMENT

If required, rapid movements can be executed without a stop. The function G28 and parameter I4= are used to state, that rapid movements are executed with (default) or without a stop.

Refer to the G28 function RAPID TRAVERSE MOVEMENTS for additional information about G28 and I4=.

MOVEMENTS IN THE MAIN PLANE

Rapid movements in the main plane, thus the plane defined with G17, G18 or G19, are executed under full control of the linear interpolator. So a straight line is made.

POLAR COORDINATES OR ONE COORDINATE AND ANGLE

Positions in the main plane can also be programmed with polar coordinates or one coordinate and angle.

POSITIONING LOGIC

The positioning logic is a fixed sequence of axis movements depending on the active main plane and the movements along the tool axis.

When the tool is moving towards the workpiece:

	G17	G18	G19
	XY-PLANE	XZ-PLANE	YZ-PLANE
1st movement	4th axis	4th axis	4th axis
2nd movement	X and Y	X and Z	Y and Z
3rd movement	Z axis	Y axis	X axis

When the tool is moving away from the workpiece:

	G17	G18	G19
	XY-PLANE	XZ-PLANE	YZ-PLANE
1st movement	Z axis	Y axis	X axis
2nd movement	X and Y	X and Z	Y and Z
3rd movement	4th axis	4th axis	4th axis

5TH & 6TH AXES

If these axes are programmed, movements along them will occur simultaneously with the 4th axis.

TOOL IN POSITIVE DIRECTION OF TOOL AXIS (G66/G67)

With G67 the tool is pointing in the positive direction of the tool axis, which means that a movement towards the work piece is in the positive direction. The positioning logic is changed accordingly.

SWITCHING OFF POSITIONING LOGIC

Sometimes the positioning logic is not required, e.g. when moving the tool to a tool change position. If the positioning logic is switched off, all axes move simultaneously.

The positioning logic can be switched off with the G28 function and the word I5=1. Refer to G28 for details.

PARALLEL AXES

If available on the machine tool, the linear axes U, V and W, which are parallel to the main axes X, Y and Z, can be used instead of X, Y and Z. Only cartesian coordinates can be used with the U, V and W axes.

ORIENTED SPINDLE STOP (D.. M19)

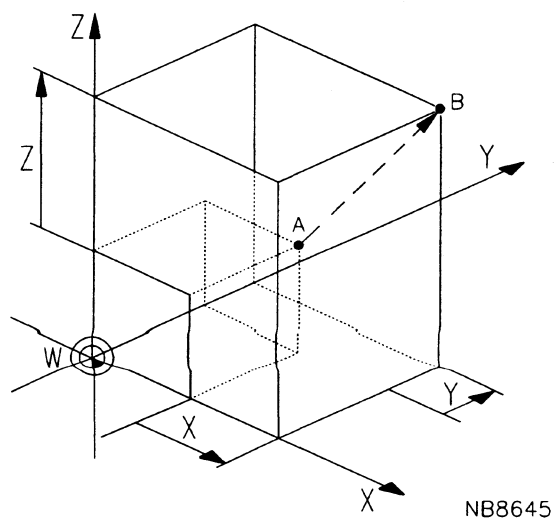
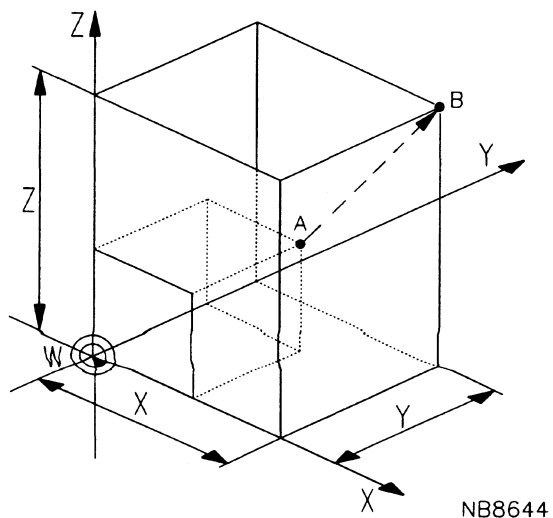
The D-word for the offset angle with oriented spindle stop must be programmed together with the function M19. Refer to the function M19 for details.

STARTING POSITION

At the beginning of a program, every active axis should be programmed in a program block for axis movements to ensure that each axis is in the starting position after a program start.

Example

EXAMPLE 1.



The rapid movement from point A to point B can be programmed as:

N... G0 X25 Y15 Z30

The actual movements are:

- a simultaneous movement in the main plane (X and Y);
- a movement in the tool axis (Z).

5. Linear interpolation G1

Purpose

Commands all linear movements to be at specified feedrates.

Format

Main plane movements

N... G1 {X..}{Y..} {Z..} {F..}

3D-interpolation

N... G1 X.. Y.. Z.. {F..}

One rotary axis

N... G1 {A..} {B..} {C..} {A40=..} {B40=..} {C40=..} {F...}

Multi axes movements

N... G1 {X..} {Y..} {Z..} {A..} {B..} {C..} {A40=..} {B40=..} {C40=..} {F..}

Parameter

End point coordinates

X,Y,Z Endpoint coordinate

A,B,C Endpoint angle

B1= Angle

L1= Path length

B2= Polar angle

L2= Polar length

P Point definition number (P1-P4)

P1= Point definition number

A40= Radius A-axis

B40= Radius B-axis

C40= Radius rotary-axis

D Angle oriented spindle stop

For absolute and incremental programming

X90=,Y90=,Z90= Absolute endpoint

A90=,B90=,C90= Absolute endpoint angle

X91=,Y91=,Z91= Incremental endpoint

A91=,B91=,C91= Incremental endpoint angle

Modal words

F,S,T,M,H,E..=

F1=, T1=, T2=

Associated functions

G0, G2/G3, G6

F-functions

Wordwise absolute/incremental programming (X90=.., X91=..) .

Type of function

Modal

Notes and Usage

AXIS MOVEMENTS

Axis movements are always interpolated, so that they occur simultaneously along all the programmed axes.

POLAR COORDINATES OR ONE COORDINATE AND ANGLE

Positions in the main plane can also be programmed with polar coordinates or one coordinate and angle.

DEFINED POINTS (G78)

A maximum of four previously defined points (P-words) can be stated in a G1 program block. The sequence in which the points are programmed also establishes the tool movement sequence e.g.: G1 P5 P2 F100

Tool moves at 100 mm/min first to P5 and then to P2.

FEED RATE IN MAIN PLANE

The programmed feed rate is the feed on the straight line.

RADIUS COMPENSATION IN MAIN PLANE (G40 - G44)

Radius compensation on contours defined with linear and circular movements is available. Refer to the functions G40, G41/G42 and G43/G44 for additional information.

3D-INTERPOLATION

If the three axes X, Y and Z are programmed in one block, the CNC will control the axes movements so that a linear movement in space is made from the start point to the point defined by the three end point coordinates.

Positions in the main plane can be programmed normally.

For programming axes outside the main plane cartesian coordinates must be used.

FEED RATE WITH 3D-INTERPOLATION

The programmed feed rate is the feed on the straight line.

3D TOOL CORRECTION

A 3D tool correction with normalized vectors is available. Refer to the G141 section for additional information.

PROGRAMMING ROTARY AXES

RADIUS OF ROTARY AXIS FOR FEED CALCULATIONS

For feed calculations the radius of each rotary axis involved can be programmed with A40= (for the A-axis), B40= (for the B-axis) and C40= (for the C-axis).

CANCELLATION OF THE RADIUS OF THE ROTARY AXIS

The programmed radius of the rotary axis is modal and, therefore, remains active until cancelled by:

- A40=0, B40=0 or C40=0.
- A different coordinate system being selected.
- M30, CLEAR CONTROL or CANCEL PROGRAM.

PROGRAMMED FEED RATE

When A40=, B40= or C40= is given a radius value, the surface feed is programmed in mm/min or inch/min.

NO RADIUS OF ROTARY AXIS PROGRAMMED

If no radii are programmed, the programmed feed rate is the feed of the path of the linear axes and is used by the CNC for calculating the feed for each rotary axis. This ensures that all axes cover the same part of their distances to go in the same time.

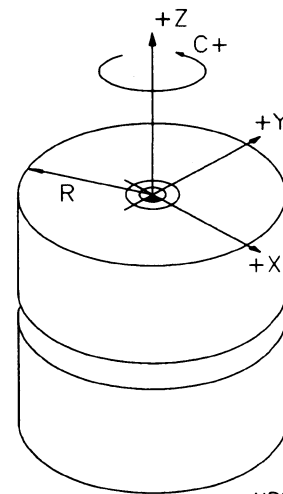
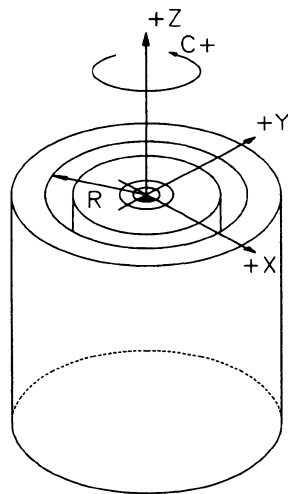
MAXIMUM FEED PER AXIS

If the maximum feed of an axis (MC-setting) is exceeded, the actual feed is reduced, so that the movement is performed with the maximum feedrate.

ONE ROTARY AXIS

Note: In the following example, the Z- and C-axis have been used, however, the same principles apply to the Y-/B-axis and X-/A-axis combinations.

With just the rotary axis moving, two cutting actions are possible.



NB8569

1. Groove cut in the facing plane

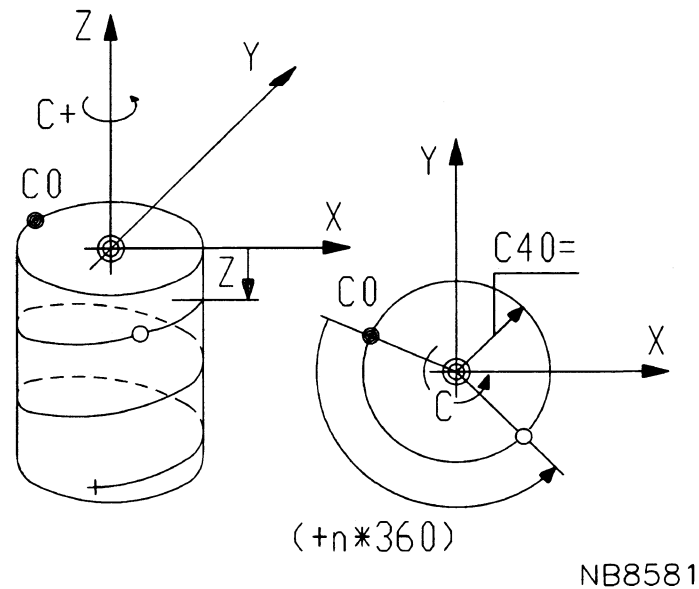
2. Groove cut in the cylinder's surface

FEED RATE WITH ROTARY AXIS ONLY

If the radius of the rotary axis is programmed, the feed rate is the surface feed in mm/min or inch/min.

If the radius of the rotary axis is not programmed or A40=0, B40=0 or C40=0 is programmed, the feed rate is the feed in degrees/min.

ONE ROTARY AXIS AND ONE LINEAR AXIS

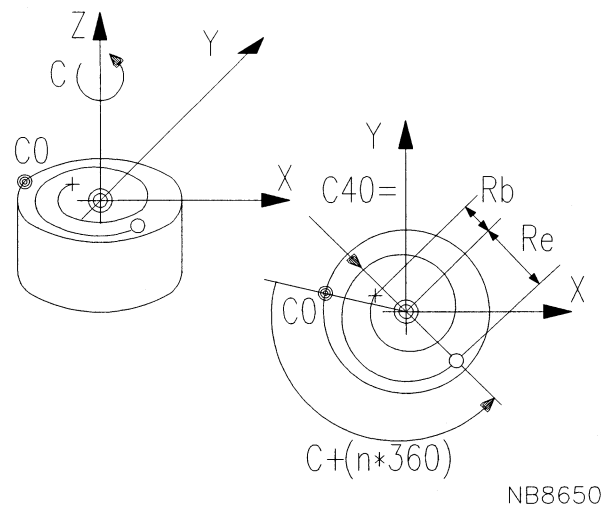


Helix on the curved surface of a cylinder

CYLINDER INTERPOLATION

If contours or positions are to be programmed on the curved surface of a cylinder, the function G182 is available for an easy programming of these movements. Refer to the G180/G182 section for additional information.

SPIRAL IN THE FACING PLANE

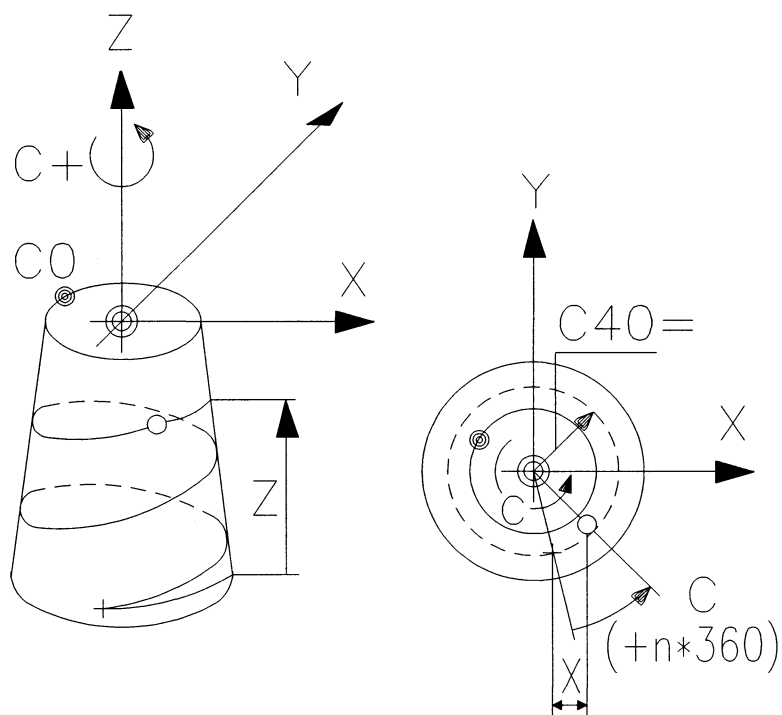


AVERAGE RADIUS OF THE SPIRAL

The radius value which should be programmed for feed calculations, is the average radius of the path; this radius is used by the CNC to calculate the required feed rate to produce the spiral. The average radius (e.g. C40=) is calculated by using the formula:

$$C40 = (\text{radius at start point}(R_b) + \text{radius at end point}(R_e)) / 2$$

ONE ROTARY AXIS AND TWO LINEAR AXES

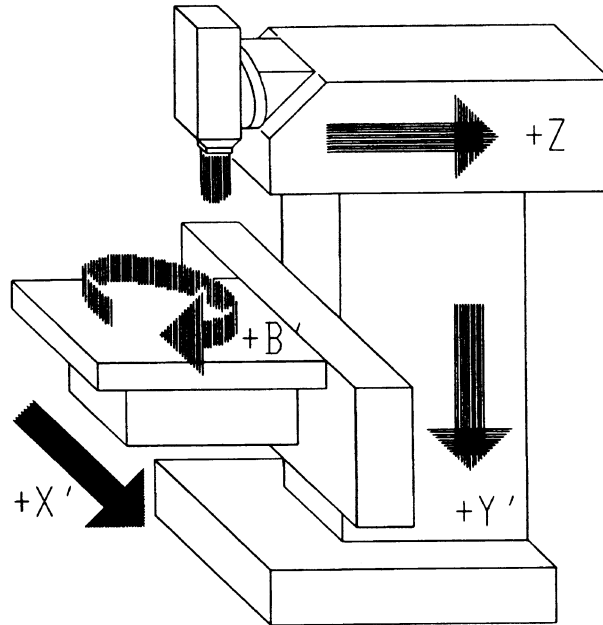


NB8833

For feed calculations the average radius (C40=) has to be used and is calculated as:

$$C40 = (\text{radius at start point} + \text{radius at end point}) / 2$$

The movement on the cone is programmed with all three axis (X.. Z.. and C..) in one block.

MULTI AXES PROGRAMMING

Machine tool with rotary table and tilting head

In a G1-block any combination of the three linear axes X, Y, or Z and the three rotary axes A, B or C (if available) is allowed.

Positions in the main plane can be programmed normally.

For programming axes outside the main plane cartesian coordinates must be used.

GENERAL REMARKS WITH G1**PARALLEL AXES**

If available on the machine tool, the linear axes U, V and W, which are parallel to the main axes X, Y and Z can be used instead of X, Y and Z. Only cartesian coordinates can be used with the U, V and W axes.

CANCELLATION

The function G1 is cancelled by functions G1, G2/G3, G6 or at end of program (M30), CLEAR CONTROL or softkey CANCEL PROGRAM.

START NEXT MOVEMENT

In general feed movements are executed without a stop between the blocks. This results in rounded corners.

G28 and parameter I3= allows to program if the next movement starts after a full stop of the tool or without a stop between the movements.

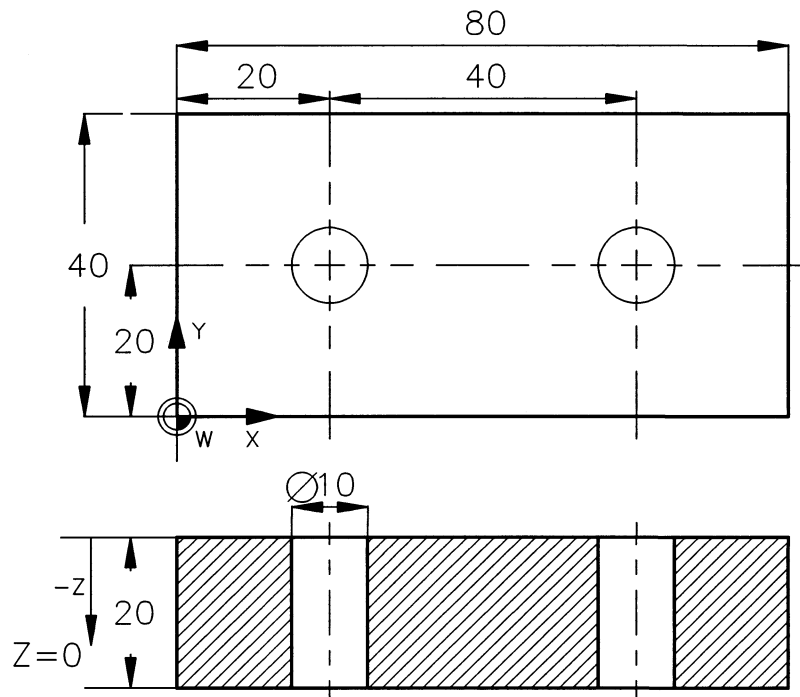
Refer to the function G28 for details.

ORIENTED SPINDLE STOP (D..M19)

The D-word for the offset angle with oriented spindle stop must be programmed together with the function M19. Refer to the function M19 for details.

Examples

EXAMPLE 1. Drilling a hole



NB8589

```

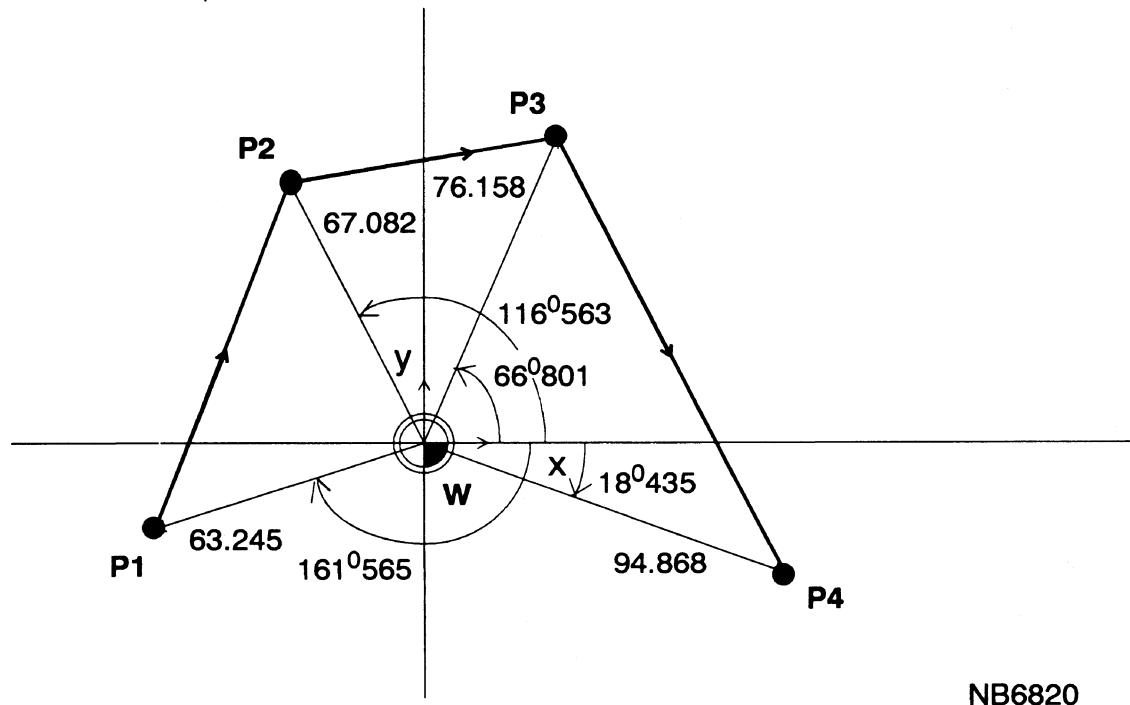
N9001
N1 G17
N2 G54 T1 M6 S1000
N3 X20 Y20 Z1 F150 M3
N4 G1 Z-23.5
N5 G0 X60 Z1
N6 G1 Z-23.5
N7 G0 Z200 M30

```

Explanation:

- N1: Activate XY-plane (G17).
 N2: Activate zero shift (G54). Load tool T1 and its offsets. Spindle rotation at 1000 rev/min. Drill diameter 10mm.
 N3: Move tool rapidly (G0) to programmed position. Set feedrate to 150 mm/min. Make spindle rotate clockwise (M3).
 N4: Feed tool to programmed depth.
 N5: Retract tool to Z1 and then move the tool rapidly to X60. The CNC's positioning logic ensures that the tool does not collide with the workpiece, because the tool is first moved along the Z-axis before moving along the X-axis.
 N6: Feed tool to programmed depth.
 N7: Retract tool to Z200 and end of program.

EXAMPLE 2. Main plane movements



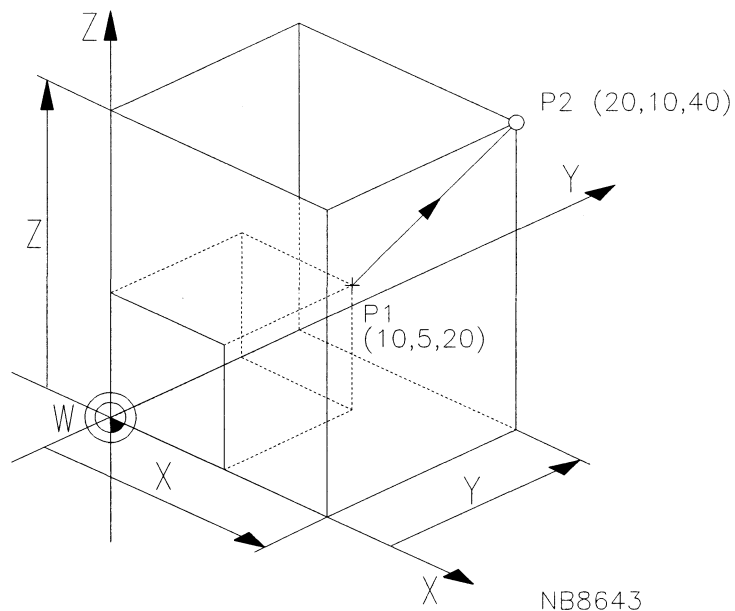
By using absolute polar coordinates the above movements can be programmed as follows:

N10 G0 B2=-161.565 L2=63.245 F100	(P1)
N11 G1 B2=116.565 L2=67.082	(P2)
N12 B2=66.801 L2=76.158	(P3)
N13 B2=-18.435 L2=94.868	(P4)

Explanation:

N10: Move at rapid traverse rate (G0) to point P1.
 N11-N13: Move tool at feedrate (100 mm/min) to the points P2, P3 and P4.

EXAMPLE 3. 3D-interpolation



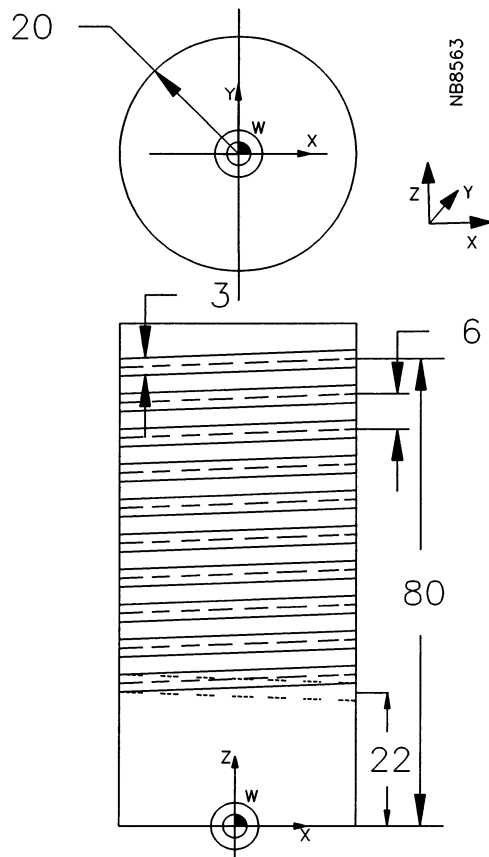
Tool moves from P1 (10,5,20) to P2 (30,10,40) at a feedrate of 100 mm/min

N14 G0 X10 Y5 Z20

N15 G1 X20 Y10 Z40 F100

In block N15 the three axes move simultaneously and reach their end positions at the same time.

EXAMPLE 4. Helix on a cylinder surface milled on a horizontal milling machine



Simultaneous movements of Z- and C-axes. The tool is in the Y-axis. The helix has 10 turns and a pitch of 6 mm.

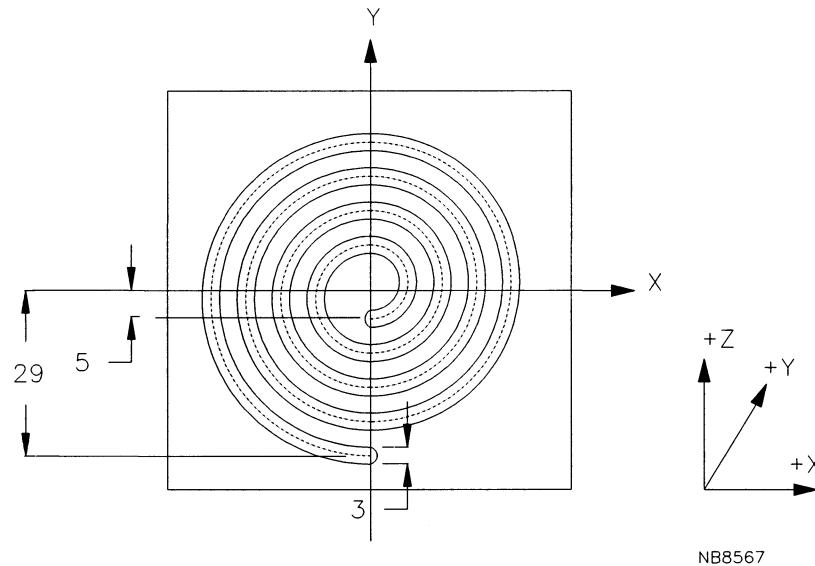
```

N10 G18
N11 T1 M6
N12 G0 X0 Y22 Z80 C0 S3000 M3
N13 G1 Y18 F75
N14 Z20 C3600 C40=20 F125
N15 G0 Y100
    
```

Explanation:

N10: Define the main plane
 N11: Load tool T1 and its offsets (mill diameter 3 mm). Tool is in Y-axis (G18).
 N12: Start spindle and move tool to start position.
 N13: Feed tool 4 mm to position Y18
 N14: Mill the helix. The rotary axis turns ten times (C3600)
 N15: Move tool away from the workpiece.

EXAMPLE 5. Spiral in the main plane (= facing plane) of a vertical milling machine.



The spiral has four turns and a pitch of 6 mm. It is produced with a rotary table (C-axis) rotating around the tool axis (Z-axis) and a simultaneous movement in the Y-axis. C40= 17

The program could look like:

```

N10 G17
N11 G54 T1 M6
N12 G0 X0 Y-5 Z2 C0 S3000 M3
N13 G1 Z-2 F75
N14 Y-29 C1440 C40=17 F200
N15 G0 Z100

```

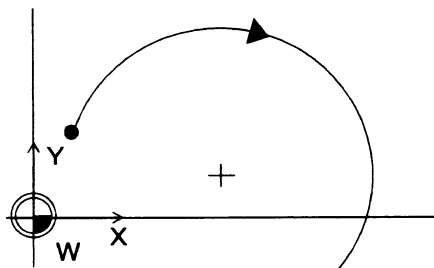
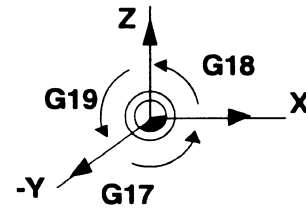
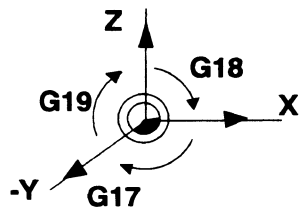
Explanation:

- N10: Define the main plane
- N11: Activate zero shift to the middle of the workpiece (G54). Load tool T1 and its offsets. The tool is in the Z-axis (G17). Mill diameter 3 mm.
- N12: Start the spindle. Move tool to start position.
- N13: Feed tool in Z-axis; 2 mm depth into the workpiece.
- N14: Mill the spiral (Rb=5, Re=29, therefore C40=17). Rotate the C-axis four times (C1440).
- N15: Move tool away from workpiece.

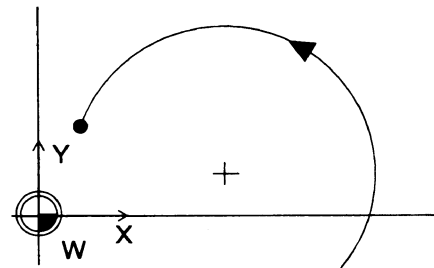
6. Circular interpolation (CW/CCW) G2/G3

Purpose

To execute a ClockWise (G2) or Counter-ClockWise (G3) circular movement at a specified feedrate.



G2



G3

NB8710

The direction of circular movement is decided by looking in the negative direction of the tool axis towards the main plane.

Format

Full circle.

N... G2/G3 [Centre point coordinates]

Arc less than or equal to 180°:

N... G2/G3 [Linear axis's end point coordinates] R...

Arc less or greater than 180°:

N... G2/G3 [Centre point coordinates] [Linear axis's end point coordinates]

N... G2/G3 [Centre point coordinates] B5=...

2.5D Interpolation

N... G2/G3 [Centre point coords] [Arc end point coords]
[Linear or rotary axis's end point coordinate]

Helix

N... G2/G3 [Centre point coords] [Arc end point coords]
[Linear or rotary axis's end point coord] [Pitch]

N... G2/G3 [Centre point coordinates] [Pitch] B5=...

Parameters

End point coordinates

X,Y,Z Endpoint coordinate

A,B,C Endpoint angle

B1= Angle

L1= Path length

B2= Polar angle

L2= Polar length

P,P1= Point definition number

Centre point coordinates

I Center point in X

J Center point in Y

K Center point in Z / pitch in Z

B3= Polar angle for center

L3= Polar length for center

Circle parameters

R Circle radius

B5= Angle of arc

D Angle oriented spindle stop

For absolute and incremental programming

X90=,Y90=,Z90= Absolute endpoint

A90=,B90=,C90= Absolute endpoint angle

I90=,K90=,J90= Absolute centre point

X91=,Y91=,Z91= Incremental endpoint

A91=,B91=,C91= Incremental endpoint angle

I91=,J91=,K91= Incremental centre point

Modal words

F,S,T,M,H,E..=

F1=, T1=, T2=

Associated functions

G0,G1,G6

F-functions

Wordwise absolute/incremental programming (X90=..., X91=..).

Type of function

Modal

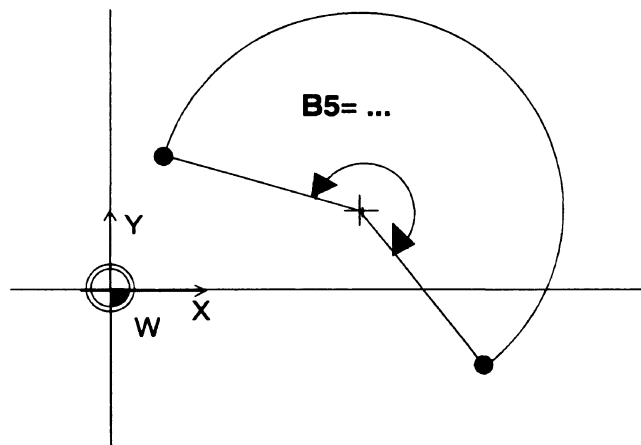
Notes and usage**CIRCLE IN THE MAIN PLANE****CIRCULAR ARC UP TO 180 DEGREES**

An arc movement up to 180 degrees is programmed by using the end point's coordinates together with the arc radius or the coordinates of the arc's centre point.

The arc radius is programmed with the R-word. This is a dimension word without a sign.

CIRCULAR ARC GREATER THAN 180 DEGREES

An arc movement greater than 180 degrees can only be programmed with the coordinates of the end point and of the arc's centre point.

ANGLE OF CIRCULAR ARC (B5=)**NB8714**

An arc of any angle between 0° and 360° can also be programmed with the centre point coordinates and the angle of the arc. The angle is programmed with the word B5= in decimal degrees and without sign.

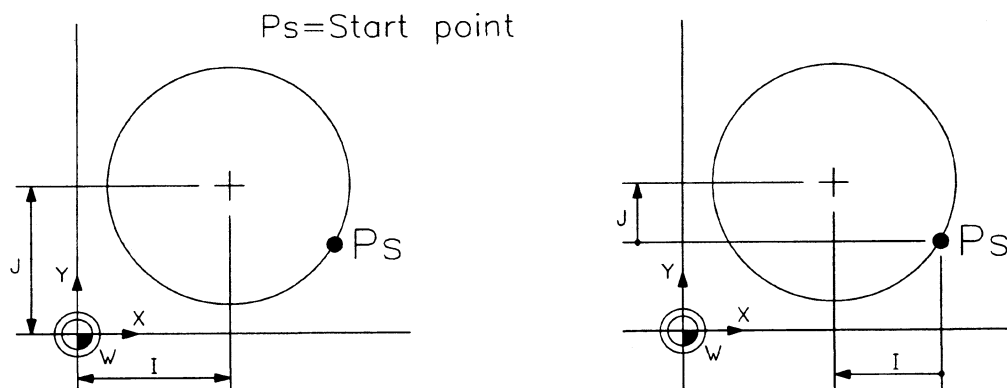
PROGRAMMING A COMPLETE CIRCLE

After making a complete circle the tool is back at its start point. This movement is programmed with:

- the direction of movement on the circle,
- the coordinates of the centre point.

Note: If the end point coordinates are also programmed in the block, no circular movement will be executed.

CENTRE POINT COORDINATES



NB8834a

Absolute (G90) coordinates

Incremental (G91) coordinates

Absolute centre point coordinates are related to the program zero point W.

Incremental centre point coordinates are measured from start point to centre point.

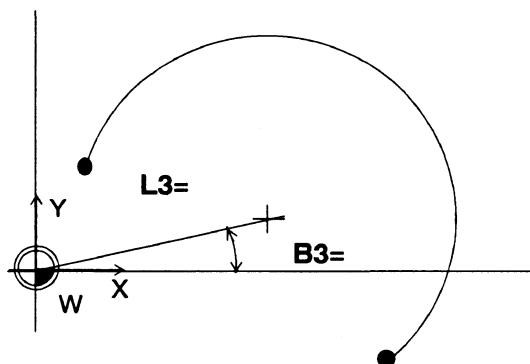
CENTRE POINT COORDINATES IN DIFFERENT PLANES

N... G2/G3 I... (X-axis) J... (Y-axis): XY-PLANE (G17)

N... G2/G3 I... (X-axis) K... (Z-axis): XZ-PLANE (G18)

N... G2/G3 J... (Y-axis) K... (Z-axis): YZ-PLANE (G19)

POLAR CENTRE POINT COORDINATES

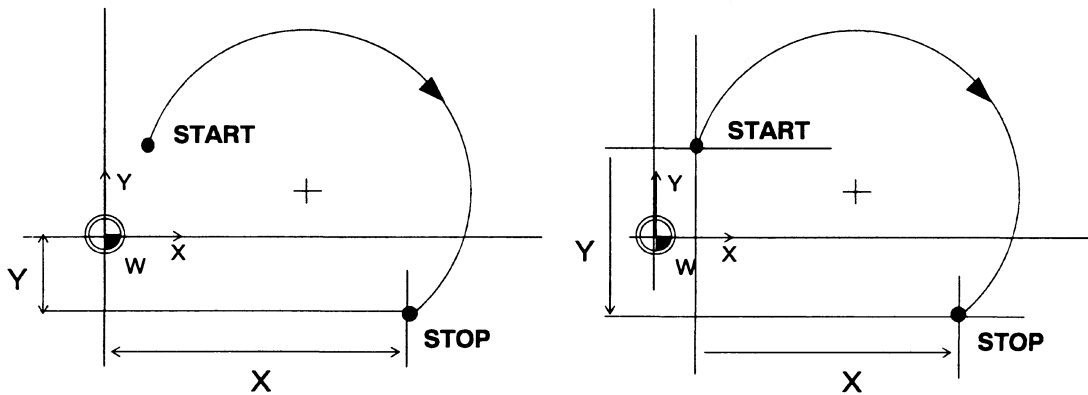


NB8711

L3=... B3=

The polar centre point coordinates are used in the plane defined by G17, G18 or G19.

CARTESIAN END POINT COORDINATES



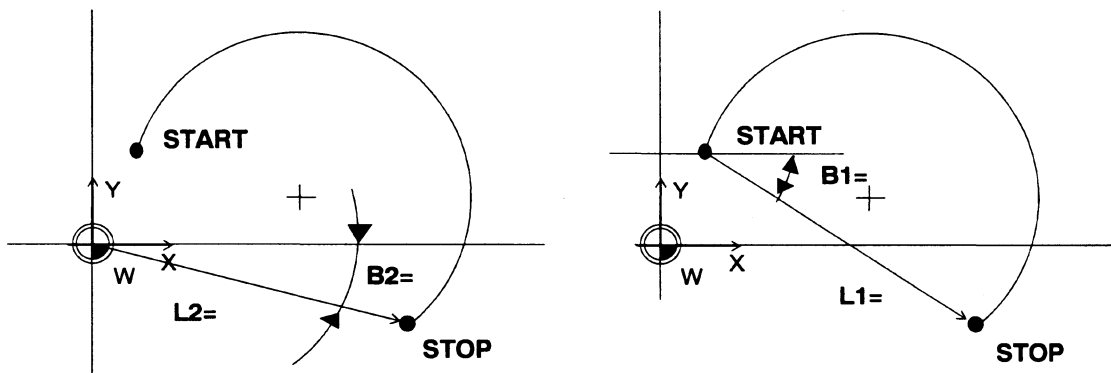
NB8712

Absolute (G90) coordinates

Incremental (G91) coordinates

(G17) X... Y... (G18) X... Z... (G19) Y... Z...

POLAR END POINT COORDINATES AND ONE COORDINATE AND ANGLE



NB8713

B2=... L2=...

B1=... L1=...

These end point coordinates are used in the plane defined by G17, G18 or G19.

DEFINED POINT (G78)

A previously defined point (P-word) can be used to program the end point of a circular movement.

FEED RATE IN MAIN PLANE

The programmed feed rate is the feed on the circle.

RADIUS COMPENSATION IN MAIN PLANE (G40 - G44)

Radius compensation on contours defined with linear and circular movements in the main plane is available. Refer to the functions G40, G41/G42 and G43/G44 for additional information.

A correction of the feed rate with circular movements and radius compensation depending on the shape of the contour and the radius of the mill, is available. Refer to **CONSTANT CUTTING FEED** with the function G41/G42 for additional information.

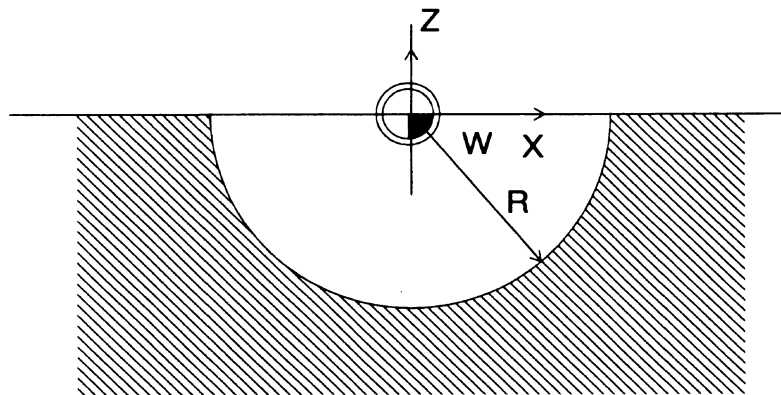
AXIS ROTATION (G92/G93 B4=)

A circular movement in a rotated main plane can be used.

A CIRCULAR MOVEMENT NOT IN THE MAIN PLANE

It is possible for a special tool such as a ball cutter, to be controlled so that it cuts in a direction being not parallel to the main plane. In these circumstances only cartesian absolute or incremental dimensions can be used to program the end point and centre point coordinates.

Radius compensation is not available.



NB8574

Circular movement in XZ-plane; tool in Z-axis

CIRCULAR ARC UP TO 180°

An arc movement up to 180 degrees is programmed by using the end point's coordinates together with the arc radius (R-word) or the cartesian coordinates of the arc's centre point.

CIRCULAR ARC GREATER THAN 180°

An arc movement greater than 180 degrees can only be programmed with the cartesian coordinates of the end point and the arc's centre point.

ADDRESSES FOR END POINT AND CENTRE POINT

The addresses given in the table under the heading CENTRE POINT COORDINATES IN DIFFERENT PLANES also apply to arc movements which are not in the current main plane. The addresses of the centre point define the plane in which the circle is to be milled.

AXIS ROTATION (G92/G93 B4=)

A circular arc is not in the plane defined by the active G-function for plane selection (G17, G18 or G19) is not allowed in a plane of which one axis is rotated. E.g. if G17 is active and the XY-plane rotated 30°, the X-axis is rotated. So a circular movement in the XZ-plane is not allowed. An error message is displayed to this effect.

LINEAR AXES U, V AND W

If a machine tool is equipped with linear axes parallel to the main axes, circular movements can be used with these axes.

Polar coordinates or radius compensation cannot be used.

PROGRAMMING A CIRCULAR ARC

For an arc movement up to 180 degrees, either the radius (R-word) or the cartesian centre point coordinates (absolute or incremental) can be programmed.

For an arc movement greater than 180 degrees only cartesian coordinates can be used for programming the centre point.

PROGRAMMING END POINT AND CENTRE POINT

Both coordinates of the endpoint must be programmed because they determine the plane in which the circular movement occurs.

Centre point coordinates are defined by: I for the U-axis; J for the V-axis; K for the W-axis.

The tables below indicate which addresses are used with different planes.

Main axis X and a linear axis

	XV-plane	XW-plane
End point	X and V	X and W
Centre point	I and J	I and K

Main axis Y and a linear axis

	YU-plane	YW-plane
End point	Y and U	Y and W
Centre point	J and I	J and K

Main axis Z and a linear axis

	ZU-plane	ZV-plane
End point	Z and X	Z and Y
Centre point	K and I	K and J

Combination of linear axes

	UV-plane	UW-plane	VW-plane
End point	U and V	U and W	V and W
Centre point	I and J	I and K	J and K

A CIRCULAR MOVEMENT WITH A SIMULTANEOUS MOVEMENT IN A THIRD AXIS

The CNC control can use a special interpolation procedure (2.5D), to coordinate a circular movement in a main plane and a third axis' movement, so that a tool travels the correct paths from the start point to the end point.

CIRCLE IN THE MAIN PLANE

The normal programming methods for a circle in the main plane defined by G17, G18 or G19 are used.

Radius compensation with the circular movement can be used. The tool must tangentially enter and leave the workpiece.

If the third axis is the tool axis, it is programmed by using one of the addresses given in the table below.

	G17	G18	G19
Plane	XY-plane	XZ-plane	YZ-plane
Tool axis	Z	Y	X

CIRCLE NOT IN THE MAIN PLANE

When the circular movement is not in the main plane, the rules given in A CIRCULAR MOVEMENT NOT IN THE MAIN PLANE must be used for programming the movement. The plane is defined by the cartesian coordinates of the centre point. An arc radius (R-word) cannot be used.

The table below lists the addresses which are used for different planes.

	XY-plane	XZ-plane	YZ-plane
End point	X and Y	X and Z	Y and Z
Centre point	I and J	I and K	J and K
Tool axis	Z	Y	X

THIRD AXIS IS A ROTARY AXIS

If the circular movement is executed in the main plane or in another plane as well, the third axis is not restricted to the tool axis, but a rotary axis programmed with the address A or B can also be used. In this case a simultaneous movement of the linear axes performing the circular movement in the defined plane, and the rotary axis occur.

HELIX INTERPOLATION

A helix on any cylinder surface can be milled by programming the following:

- circular movement in the main plane as described
- the pitch of the helix
- (if necessary) the end point of the linear movement.

	G17	G18	G19
Tool axis	Z	Y	X
Centre point	I and J or B3= and L3=	I and K or B3= and L3=	J and K or B3= and L3=
Angle of arc	B5=	B5=	B5=
Pitch of helix	K	J	I

PROGRAMMING THE ARC ANGLE

The value of 'B5=' can be from 0 to 999999 degrees, which is approximately 900 revolutions.

PROGRAMMING THE TOOL AXIS

A helix movement can also be programmed by using the addresses given in the table below.

	G17	G18	G19
Tool axis	Z	Y	X
Circle end point	X and Y	X and Z	Y and Z
Centre point	I and J	I and K	J and K
Pitch of helix	K	J	I

When these alternative addresses are used, the movements have to be programmed so that the circular movement and the tool axis movement reach their end positions at the same time.

THIRD AXIS IS A ROTARY AXIS

The third axis is not restricted to the tool axis, but a rotary axis programmed with the address A or B can also be used. In this case a simultaneous movement of the linear axes performing the circular movement, and the rotary axis occur.

RADIUS COMPENSATION WITH HELIX INTERPOLATION

Radius compensation with the circular movement can be used during helix interpolation. The tool must tangentially enter and leave the workpiece.

GENERAL REMARKS WITH G2/G3

CANCELLATION

The function G2 or G3 is cancelled by any other function of group A, or at end of program (M30), or by CLEAR CONTROL.

CHECKING OF CENTRE POINT COORDINATES

When centre point coordinates are used, the radius of the circular movement at the start is compared with the radius at the end. If the difference between the two values is greater than a Machine Constant setting, the CNC generates an error message and stops program execution.

START NEXT MOVEMENT (G28 I3=..)

In general feed movements are executed without a stop between the blocks. This results in rounded corners.

G28 and parameter I3= allows to program if the next movement starts after a full stop of the tool or without a stop between the movements.

Refer to the function G28 for details.

FEED LIMITATION (G28 I6=..)

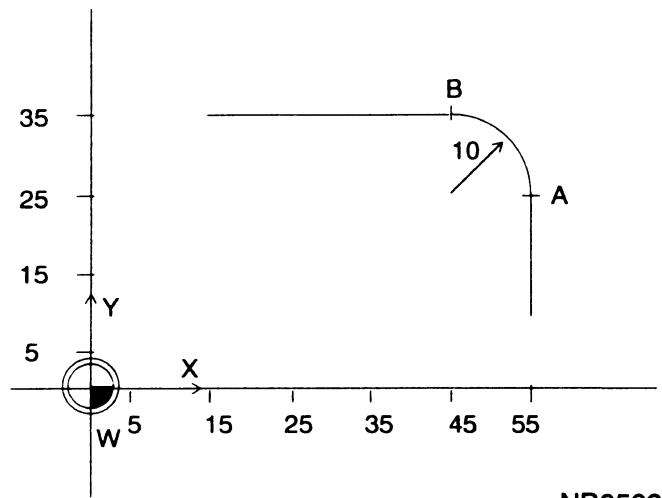
With feed limitation the programmed feed rate is reduced to keep the axes following error to an acceptable minimum and therefore to improve the machining accuracy. Refer to the function G28 I6= for details.

ORIENTED SPINDLE STOP (D.. M19)

The D-word for the offset angle with oriented spindle stop must be programmed together with the function M19. Refer to the function M19 for details.

Note: If a circle is programmed and one of its endpoints is within some microns from the startpoint, then at high feed values the cnc will not perform a circle but a linear movement directly to the endposition.

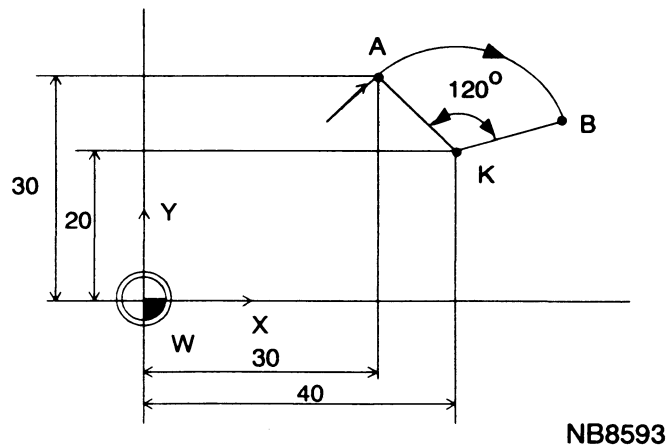
Example: if difference between start and end position is 5 micron, then if the feed exceeds 1 m/ min then a linear movement will be executed although G2/G3 has been programmed.

Examples**EXAMPLE 1. Programming an arc radius**

N10 G1 X55 Y25 F100 (A)
 N20 G3 X45 Y35 R10 (B)

Explanation:

N10: Move tool at set feedrate to the starting point A of the arc.
 N20: Move tool in a counter-clockwise (G3) direction to end point B.

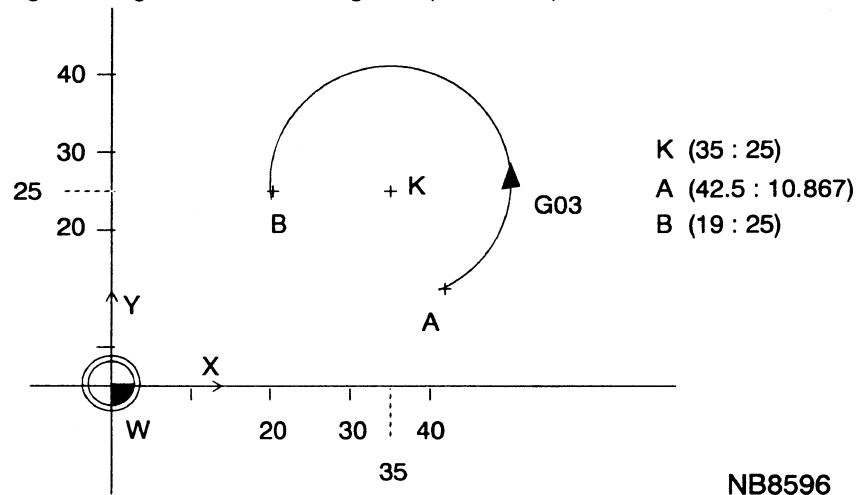
EXAMPLE 2. Programming an arc angle.

N10 G1 X30 Y30 F500 (A)
 N11 G2 140 J20 B5=120 (B)

Explanation:

N10: Move tool at set feedrate to starting point A.
 N11: Move tool in a clockwise direction (G2) to end point; the angle of the arc is stated by B5=.

EXAMPLE 3. Programming an arc > 180 degrees (Cartesian)



Absolute coordinates

```
N10 G1 X42.5 Y10.867 F200          (A)
N11 G3 X19 Y25 I35 J25             (B)
```

Explanation:

- N10: Move tool at given feedrate to the starting point A of the arc.
 N11: Move tool in a counter-clockwise (G3) direction to end point B. Centre point coordinates are stated by I and J.
 Both the coordinates X and Y as well as I and J are absolute values with regard to the program zero point W.

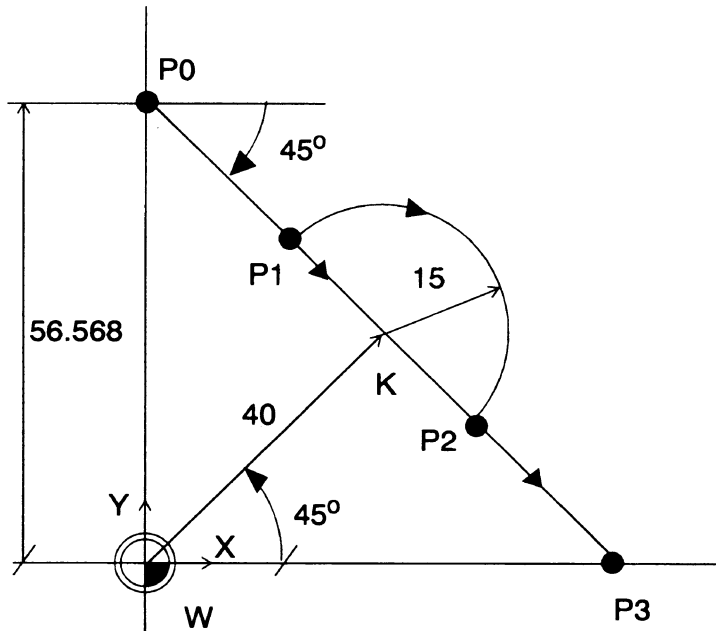
Incremental coordinates

```
N10 G1 X42.5 Y10.867 F200          (A)
N11 G91
N12 G3 X-23.5 Y14.133 I-7.5 J14.133 (B)
```

Explanation:

- N10: Move tool at given feedrate to the starting point A of the arc.
 N11: Activate incremental coordinate mode (G91).
 N12: Move tool in a counter-clockwise (G3) direction to end point B. Coordinates X and Y are increments from point A to B. The coordinates I and J are incremental values from A to the centre.

EXAMPLE 4. Programming an arc with polar coordinates



NB8576

```

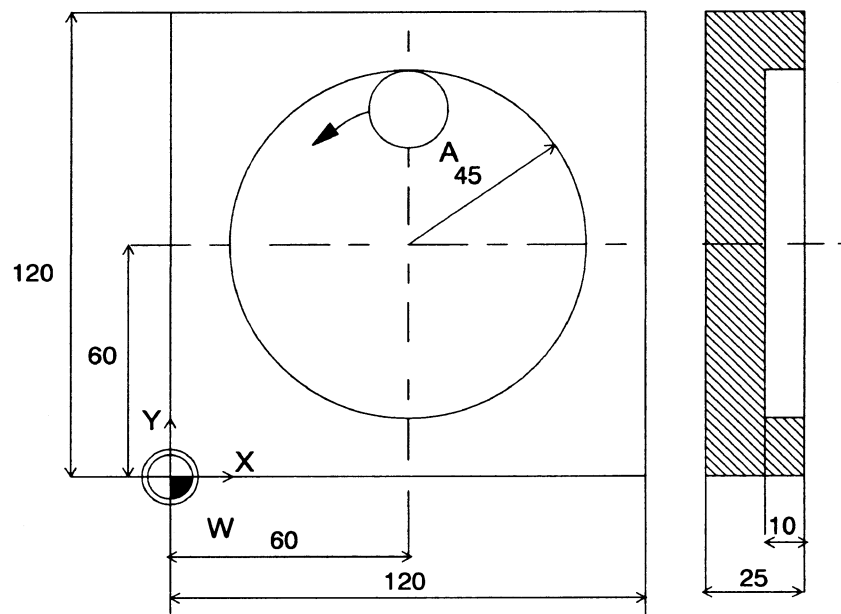
N10 G0 X0 Y56.568          (P0)
N11 G1 B1 =-45 L1=25 F200  (P1)
N12 G2 B1 =-45 L1=30 B3=45 L3=40 (P2)
N13 G1 B1 =-45 L1=25          (P3)

```

Explanation:

- N10: Move tool rapidly to point P0.
- N11: Move tool at programmed feedrate from P0 and P1. Incremental polar coordinates are used.
- N12: Move tool in a clockwise (G2) direction from P1 to P2. For P2, incremental coordinates are used. The centre point is programmed with absolute polar coordinates (B3=,L3=).
- N13: Move tool at set feedrate from P2 to P3.

EXAMPLE 5. Programming a complete circular movement.



NB8575

```

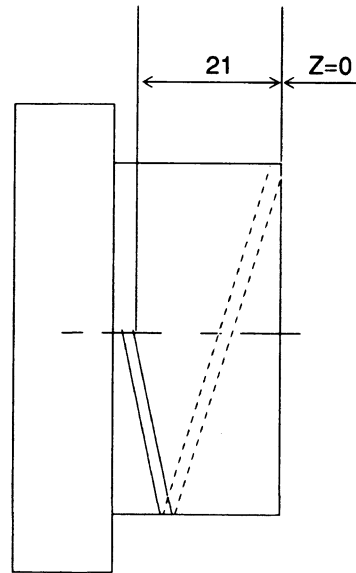
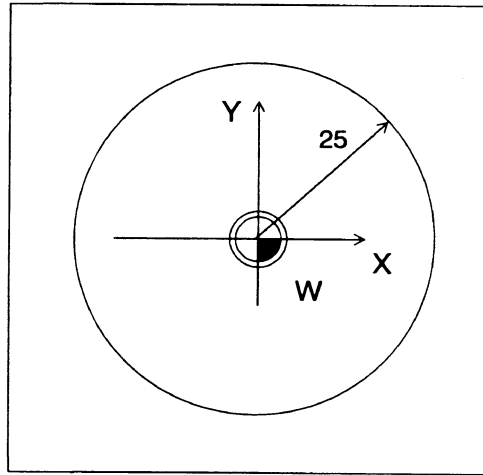
N9 G17 T1 M6
N10 G0 X60 Y60 Z10 S315 M3
N11 G1 Z-10 F36
N12 G43 Y65
N13 G41
N14 G3 I60 J60
N15 G40
N16 G1 Y60
N17 Z100

```

Explanation:

- N9: Activate XY-plane. Load tool T1 and its offset. (Mill diameter 16 mm)
 N10: Start the spindle and move tool rapidly to the centre of the pocket (Z10) (where the tool is to enter the hole). The feedrate is 36 mm/min
 N11: Set the feedrate to 65 mm/min and feed tool depth.
 N12: Move tool with feed to the wall (G43). To point A
 N13: Set radius compensation for a tool moving on the left handside (G41).
 N14: Mill the complete circle in a counter clockwise direction (G3).
 N15: Cancel radius compensation (G40).
 N16: Move tool away from milled surface of workpiece.
 N17: Retract tool out of workpiece.

EXAMPLE 6. Programming a circular movement together with a simultaneous linear axis movement (2.5D Interpolation).



NB8577

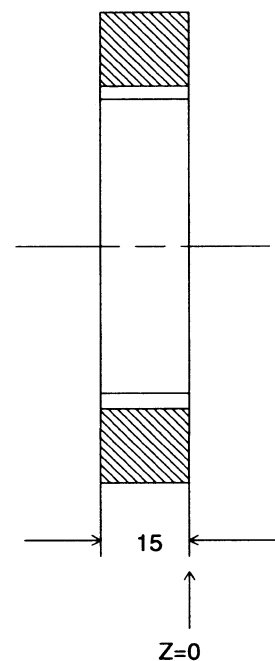
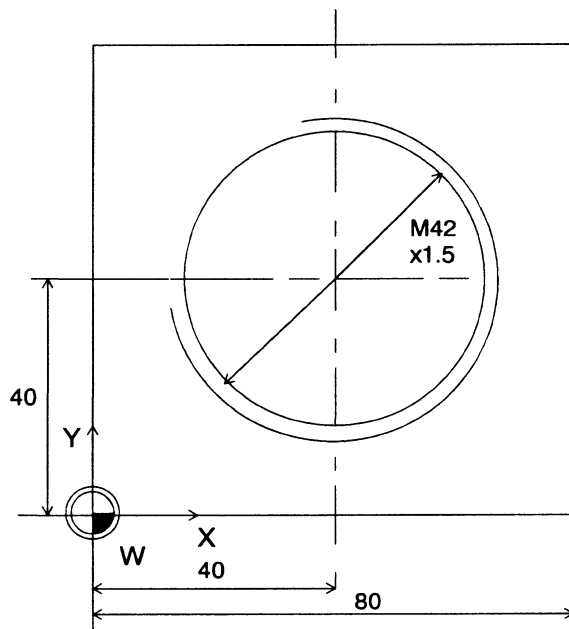
```

N10 G17 T1 M6
N11 G0 X0 Y35 Z0 S3000 M3
N12 G43
N13 G1 Y25 F80
N14 G41 F120
N15 G2 X-25 Y0 Z-21 I0 J0
N16 G40
N17 G1 X-35
  
```

Explanation:

N10: Activate XY-plane. Select tool 1 (diameter 3 mm) and its offsets.
 N11: Start the spindle and move the tool to the workpiece at 3000 rev/min
 N12: Activate radius compensation to the endpoint
 N13: Move tool to the workpiece contour. Set linear feedrate to 80 mm/min.
 N14: Set radius compensation LEFT.
 N15: Perform the simultaneous movement of the circle and third axis.
 N16: Cancel radius compensation.
 N17: Move tool away from workpiece.

EXAMPLE 7. Programming a helix.



```

N10 G17
N11 T1 M6
N12 G0 X40 Y40 Z1.5 S400 M3
N13 G1
N14 G43 Y61 F120
N15 G42
N16 G2 I40 J40 K1.5 B5=4320
N17 G40
N18 G1 Y40
N19 G0 Z100

```

Explanation:

N10: Define the main plane
 N11: Load tool 1 and its offsets. Spindle rotation 400 rev/min
 N12: Start the spindle and move tool to starting position.
 N13: Set linear feed movement
 N14: Move tool with feed to (G43) the part.
 N15: Set radius compensation RIGHT (G42).
 N16: Mill the helix. Programmed arc:
 - circle centre (I and J)
 - angle of the arc (B5=) 12 turns of 360 degrees
 - pitch of thread (K).
 N17: Cancel radius compensation (G40).
 N18: Move tool away from the wall.
 N19: Retract the tool.

By using an alternative set of addresses, block 16 could be re-written as:

```
N16 G2 X40 Y61 Z-16.5 140 J40 K1.5
```

Programmed arc:

- circle end point (X and Y)
- depth (Z)
- circle centre (I and J)
- pitch of thread (K).

7. Dwell time G4

Purpose

To insert a dwell period in the execution of a program.

Format

N... G4 X...

Minimum dwell period: 0.1 second.

Maximum dwell period: 900 seconds (15 minutes).

Parameters

X Dwell time in sec. (0.1-983)

Modal parameters

S, T, M, H, E..=

T1=, T2=

Type of function

Non-modal

Notes and usage

MODAL PARAMETERS

It is advised to program this function in a separate block without modal parameters.

If modal parameters are used in a dwell block, they are executed before the dwell.

Those M-functions which are executed after the commands in a block, are also executed after the dwell.

EXECUTION

After the dwell the machining is continued as normal.

Example

N50 G4 X2.5

The above block causes a dwell of 2.5 seconds between two operations.

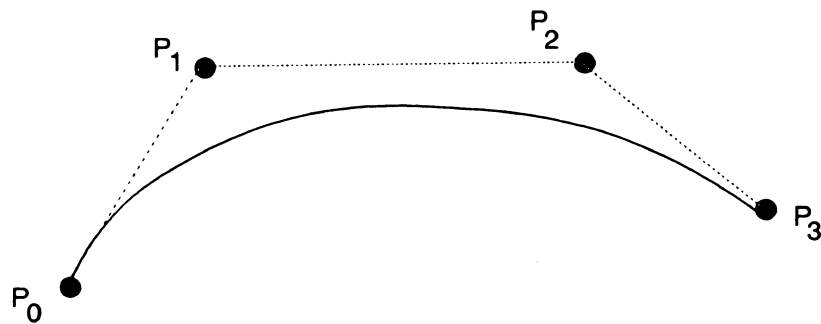
8. Spline interpolation G6

Purpose

The spline interpolation of the CNC enables the part programmer to input a series of points and have the control fit a smoothly faired curve through them.

By using this function, machine dynamic response is improved and leads to smoother tool movements and improved machining accuracy.

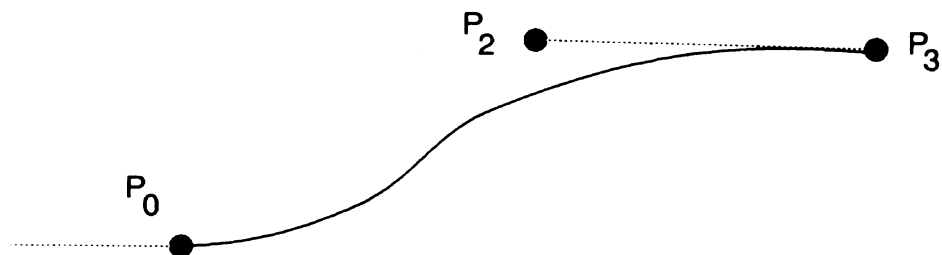
Formats with Bezier splines



NB9771

Spline with three vertex points

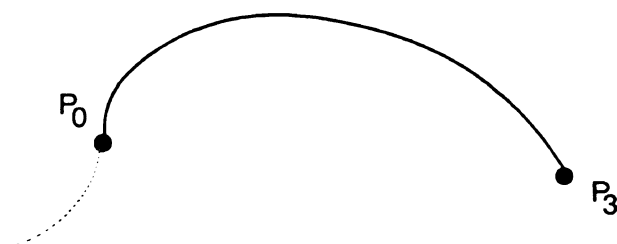
G6 X61=.. Y61=.. Z61=.. X62=.. Y62=.. Z62=.. X.. Y.. Z..



NB9772

Spline with two vertex points and constant tangent with previous spline

G6 X62=.. Y62=.. Z62=.. X.. Y.. Z..



NB9769

Spline with constant curvature with previous spline

G6 X.. Y.. Z..

Formats with cubic splines

Spline with all coefficients defined

G6 X51=.. Y51=.. Z51=.. X52=.. Y52=.. Z52=.. X53=.. Y53=.. Z53=..

Spline with constant tangent with previous spline

G6 X52=.. Y52=.. Z52=.. X53=.. Y53=.. Z53=..

Spline with constant curvature with previous spline

G6 X53=.. Y53=.. Z53=..

Parameters

X Endpoint (X-axis)

Y Endpoint (Y-axis)

Z Endpoint (Z-axis)

X61= First support point (X-axis)

Y61= First support point (Y-axis)

Z61= First support point (Z-axis)

X62= Second support point (X-axis)

Y62= Second support point (Y-axis)

Z62= Second support point (Z-axis)

Cubic Splines

X51=,Y51=,Z51= First spline coefficient

X52=,Y52=,Z52= Second spline coefficient

X53=,Y53=,Z53= Third spline coefficient

Modal functions

F,M

Associated functions

G0, G1, G2/G3

F-functions

Type of function

modal

Notes and usage with Bezier splines**DEFINITION BEZIER SPLINES**

A bezier spline is a spline defined by four points, the so called vertex points.

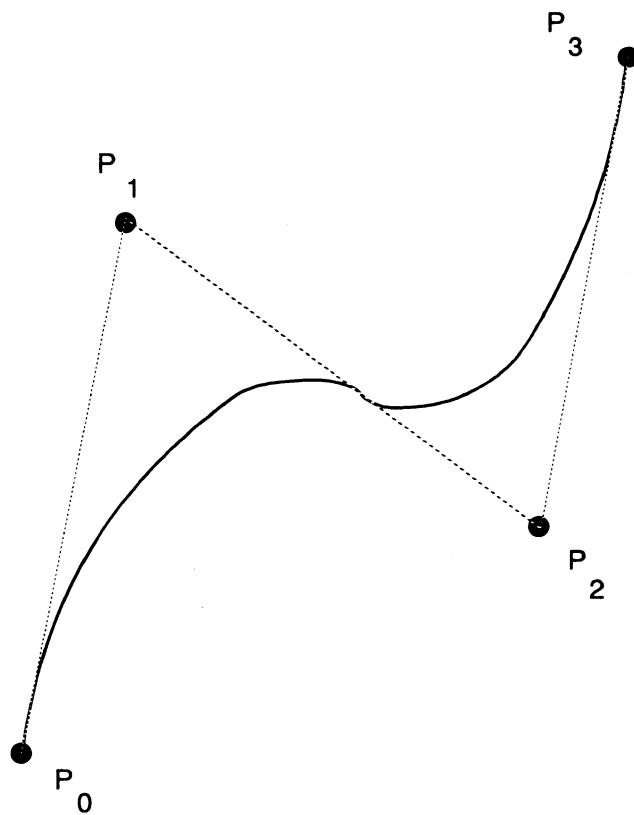
The first point of the spline is the end point of the previous movement. The curve passes through this point.

The first and second vertex point control the shape of the spline. The curve does not pass through these points.

The last point is the end point of the spline and this is, like the first point, a point through which the curve passes.

COORDINATES OF THE VERTEX POINTS

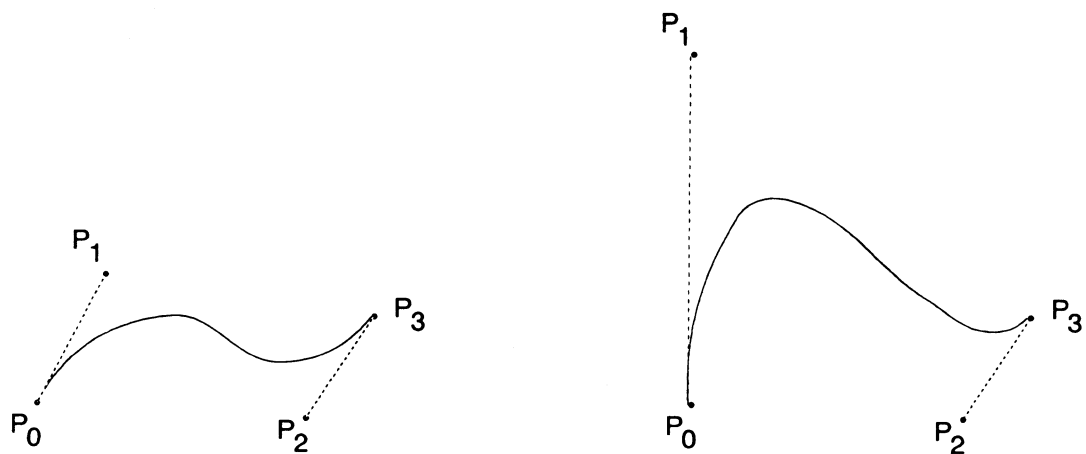
Only absolute cartesian coordinates of the main axes X, Y and Z can be used for programming the vertex points. The coordinates are related to the program zero point. The coordinates of all three axes must be entered, even if not changed.

TANGENT AT START OF SPLINE

NB9764

From the coordinates of the first vertex point (X61=, Y61=, Z61=) the tangent at the start of the spline is calculated as the line through the end point of the previous spline and the first vertex point.

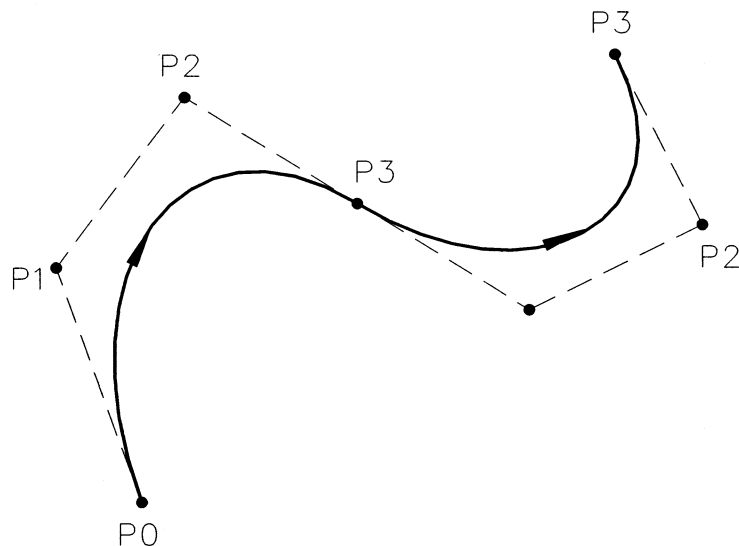
CHANGING THE LOCATION OF THE FIRST VERTEX POINT



NB9765

In the illustration three points of the spline are fixed and just the location of the first vertex point is changed. Notice that the shape of the curve changes.

TANGENTIAL CONTINUITY AT START OF SPLINE



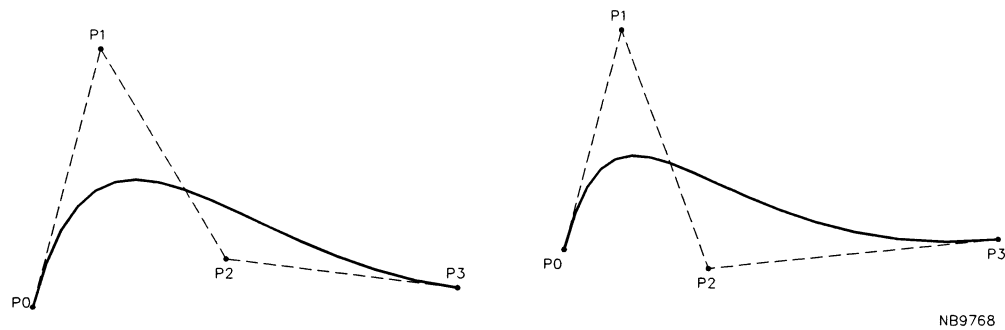
NB9766

If the first vertex point is not programmed, the tangent at the end of the previous spline is used as the tangent at the start of the spline.

TANGENT AT END OF SPLINE

From the coordinates of the second vertex point (X62=, Y62=, Z62=) the tangent at the end of the spline is calculated as the line through the second vertex point and the end point.

CHANGING THE LOCATION OF THE SECOND VERTEX POINT



In the illustration three points of the spline are fixed and just the location of the second vertex point is changed. Notice that the shape of the curve is influenced by this change in location.

CONSTANT CURVATURE

If the first and second vertex point are not programmed, constant curvature between the splines is assumed.

GETTING STARTED WITH A GROUP OF BEZIER SPLINES

If a curve or surface is defined by a group of G6-blocks with Bezier splines, the first spline must be programmed with three vertex points.

One way of doing this is to program a spline with three equal vertex points. These points are on a straight line which is tangent to the spline in the end point of the "line".

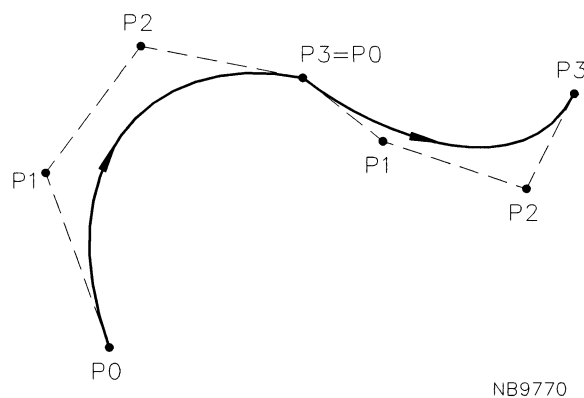
Example: N10 G1 X10 Y10 Z10
 N11 G6 X61=25 Y61=25 Z61=10 X62=25 Y62=25 Z62=10 X25 Y25
 Z10

The block N11 generates a straight line in the XY-plane.

CONNECTING SPLINES

The three different types of Bezier splines can be connected to form a curve or surface.

CORNER BETWEEN CURVES



If a corner between the splines, the second spline must be programmed with three vertex points.

RADIUS COMPENSATION

Radius compensation on splines is not available, so the tool path must be programmed when using splines.

CONNECTION BETWEEN A LINE OR CIRCLE AND A BEZIER SPLINE

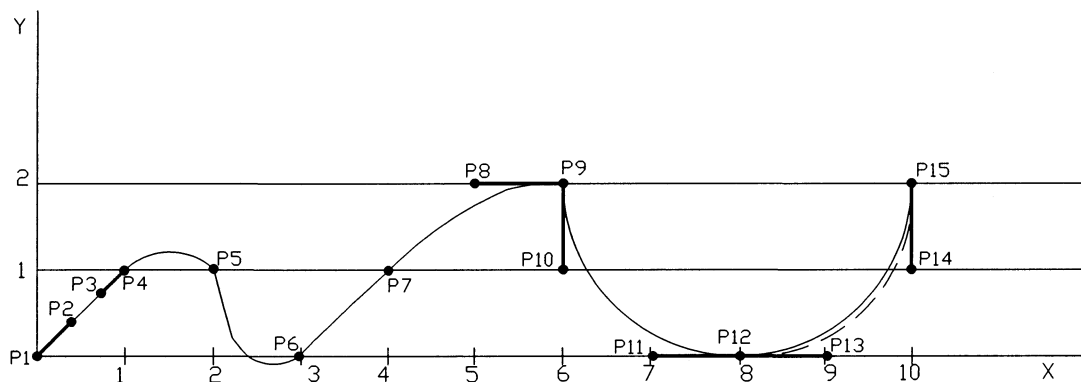
If the previous movement is linear or circular, its end point can be used as the start point of a Bezier spline with three vertex points.

Because radius compensation with splines is not available, the end point of the line or circle must also be programmed without radius compensation.

ABSOLUTE AND INCREMENTAL PROGRAMMING (G90/G91)

If G91 is active during a section with G6-blocks, the incremental programming is ignored with the G6-blocks and executed with the linear or circular movements.

It is advised to use always absolute programming (G90 active), when splines are involved.

Example:

```

N17001 (Spline curve)
N1 G98 X2 Y-6 Z-2 I10 J10 K10
N2 G17
N101 G0 X0 Y0 Z0 F500
N102 G6 X1 X61=0.3 X62=0.7 Y1 Y61=0.3 Y62=0.7 Z0.001 Z61=0 Z62=0
N103 X2 Y1.001 Z0
N104 X3 Y0 Z0.001
N105 X4 Y1 Z0
N106 X6 X62=5.7 Y2 Y62=2 Z0.001 Z62=0
N107 X8 X61=6 X62=7.5 Y0 Y61=1.5 Y62=0 Z0 Z61=0 Z62=0.001
N108 X10 X61=8.5 X62=10 Y2 Y61=0 Y62=1.5 Z0.001 Z61=0.001 Z62=0
N109 G0 X0 Y0 Z0
N110 M30

```

Explanation:

N101: Approaching starting position of curve (P1)
 N102: First curve element. Straight line. Touches P1-P2 and P3-P4. End point is P4. All coordinates should be entered. Select a straight line.
 N103: Curve passes through P5
 N104: Curve passes through P6
 N105: Curve passes through P7
 More points should be added if the curve differs from the required shape.

- N106: Curve passes through P9 and touches line P8-P9.
- N107: New curve with sharp transition is defined. First curve element starts in P9 and touches P9-P10 and P11-P12. End point is P12.
- N108: New curve with tangential transition is defined. First curve element starts in P12 and touches P12-P13 and P14-P15. End point is P15. The radius of curvature may be adjusted in P15 by changing distance P14-P15.
- N109: Return to starting position.

Note: In G6, the same coordinates should be different in two blocks (Z0 and Z0.001)

Notes and usage with cubic splines

In-depth knowledge of the cubic spline coefficients is essential. Calculations are usually done by a **CAD system**. The tool path is then also generated.

POLYNOMIAL EXPRESSION FOR CUBIC SPLINE

The cubic spline is defined by a polynomial expression.

For e.g. the X-axis, the polynomial expression for the cubic spline is (t is a parameter with $0 < t < 1$):

$$X = [X53] t^3 + [X52] t^2 + [X51] t$$

For the other axes the same polynomial is used with the programmed coefficients and the same parameter t for all axes.

These three polynomial expressions define a patch of the surface in space.

DETERMINING THE COEFFICIENTS

It is not easy to calculate the coefficients of the cubic polynomial. A good knowledge of spline coefficients is required. These calculations are mostly performed by a CAD-system which also generates the tool path.

TANGENTIAL CONTINUITY AT START OF SPLINE

If the first order coefficients are not programmed, the tangent at the end of the previous spline is used as the tangent at the start of the spline. From this tangent line the missing coefficients are calculated.

CONSTANT CURVATURE

If the first and second order coefficients are not programmed, constant curvature between the splines is assumed. The missing coefficients are calculated by the control.

Note: Omitting coefficients is only possible if the missing coefficients can be calculated from previous spline blocks.

GETTING STARTED WITH A GROUP OF CUBIC SPLINES

If a curve or surface is defined by a group of G6-blocks with cubic splines, the first spline must be programmed with all three coefficients.

Notes and usage with both types of splines

MIXING BEZIER AND CUBIC SPLINES

Both types of splines can be mixed at will.

GRAPHIC SIMULATION

The splines can be displayed in synchron graphics. Other graphic modes ignore the spline function.

CANCELLATION

The function G6 is cancelled by one of the functions G0, G1, G2 or G3 at end of program (M30), CLEAR CONTROL or the softkey CANCEL PROGRAM.

PLANE SELECTION (G17/G18/G19)

Splines are independent of the selected plane. The points (coefficients) determine the plane in which the spline is made.

ZERO POINT SHIFT (G92/G93)

If a zero point shift is programmed between G6-blocks, it is ignored with the G6-blocks and used with the other blocks.

AXIS ROTATION (G92/G93 B4=..)

Axis rotation is ignored with the G6-blocks and used with the other blocks.

SCALING AND MIRROR IMAGE (G73)

Scaling and mirror image are ignored with the G6-blocks and used with the other blocks.

START NEXT MOVEMENT

In general feed movements are executed without a stop between the blocks. This results in rounded corners.

G28 and parameter I3= allows to program if the next movement starts after a full stop of the tool or without a stop between the movements.

Refer to the function G28 for details.

RESTRICTIONS

1. Corner accuracy (G28 I3=2 or 3) and feed limitations (G28 I6=) can not be used together with spline functions
2. The BTR-function can not be used with splines.
3. When geometric calculations is active (G64) the spline-interpolation can not be used.

9. Defining the pole point (size reference point) G9 (from V320)

Purpose

To program a pole point. When a pole point has been programmed, program blocks with pole programming (angle and length) no longer relate to the zero point, but to the pole point most recently programmed.

The pole point is programmed as a function of the modally valid system of measurement G90/G91. Furthermore, the pole point may be programmed wordwise absolute, incrementally or combined absolute/incrementally.

Format

G17 active: N.. G9 X.. Y.. {X90=...} {X91=...} {Y90=...} {Y91=...}
 G18 active: N.. G9 X.. Z.. {X90=...} {X91=...} {Z90=...} {Z91=...}
 G19 active: N.. G9 Y.. Z.. {Y90=...} {Y91=...} {Z90=...} {Z91=...}

Deactivate pole (identical with workpiece zero point)
 N.. G9 X0 Y0

Pole point in polar coordinates (G17, G18, G19 active):
 absolute:
 N.. G9 B2=.. L2=.. {B1=..} {L1=..}

Parameter

Endpoint coordinates
 X,Y,Z Pole coordinate
 B1= Angle
 L1= Path length
 B2= Polar angle
 L2= Polar length

For absolute and incremental programming
 X90=,Y90=,Z90= Absolute pole coordinate
 X91=,Y91=,Z91= Incremental pole coordinate

Associated functions

G0, G1, G2, G3, G11, G40, G41, G42, G43, G61, G62, G64, G77, G78, G79, G92, G93, G145,
 Wordwise absolute/incremental programming (X90=.., X91=..)

Type of function

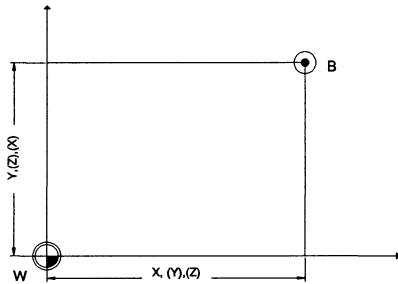
Modal

Notes and usage

Pole point in cartesian coordinates:

Pole point in absolute coordinates:

The programmed coordinates relate to the workpiece zero point.

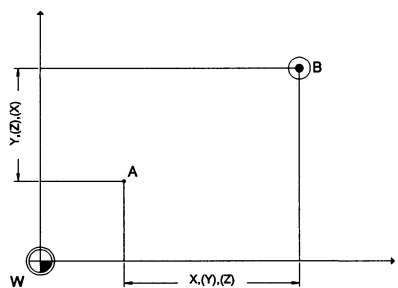


B = pole point

N.. G9 X.. Y..

Pole point in incremental coordinates:

The programmed coordinates relate to the actual position.

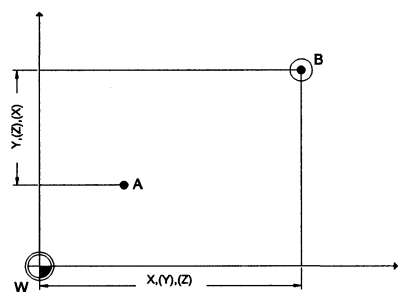


A = existing pole point

B = new pole point

N... G9 X91=... Y91=...

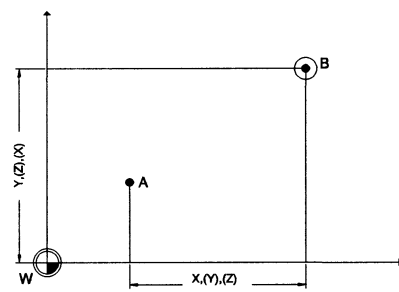
Pole point in combined absolute/incremental coordinates:



A = existing pole point

B = new pole point

N... G9 X... Y91=...

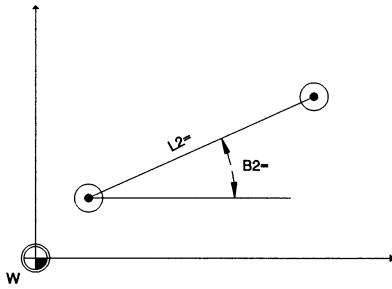


N.. G9 X91=.. Y..

Pole point in polar coordinates (G17, G18, G19 active):

Pole point in absolute polar coordinates:

The polar coordinates B2= and L2= relate to the most recently active pole point.



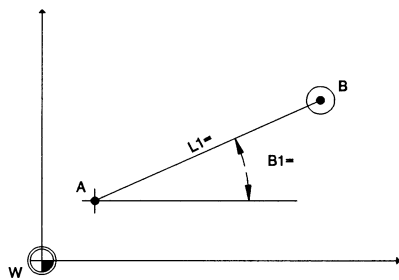
A = existing pole point

B = new pole point

N.. G9 B2=.. L2=..

Pole point in incremental polar coordinates:

The polar coordinates B1= and L1= relate to the actual position.

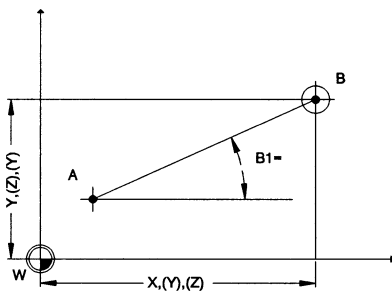


A = end point of last movement

B = new pole point

N.. G9 B1=.. L1=..

Combined programming: cartesian absolute/polar:

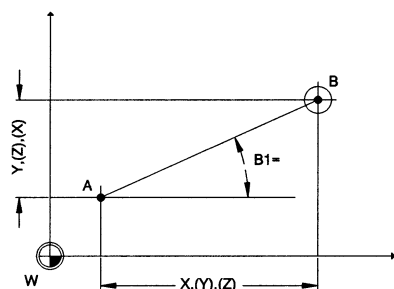


A = existing pole point

B = new pole point

N.. G9 X.. B1=..

Combined programming: cartesian absolute/polar:



A = existing pole point

B = new pole point

N.. G9 X91=.. B1=..

- polar definitions are allowed in the active working plane only
- before the G9 block is called, the pole point is at the workpiece zero point (pole point = 0)
- the pole point is modally active
- the pole point may be redefined indefinitely
- the pole point is zeroed (0) when changing the plane using G17, G18, G19

Polar definition of end point:

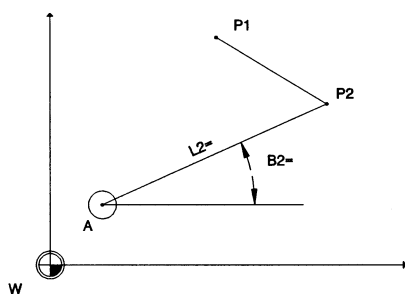
During absolute polar programming the polar lengths L2= and L3= and polar angles B2= and B3= no longer relate to the zero point, but to the pole point.

If no pole point has been defined, the pole point = 0 (zero) and therefore equals the active zero point.

Pole point definition

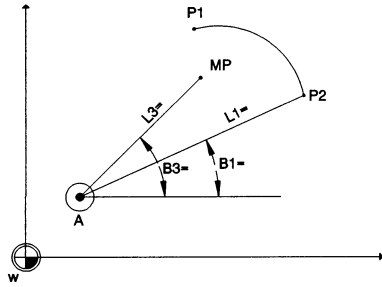
Pole points with pole can be defined in the following G-functions:

G0, G1, G40..G44, G61, G62, G77, G78, G79, G145



Polar circle definition

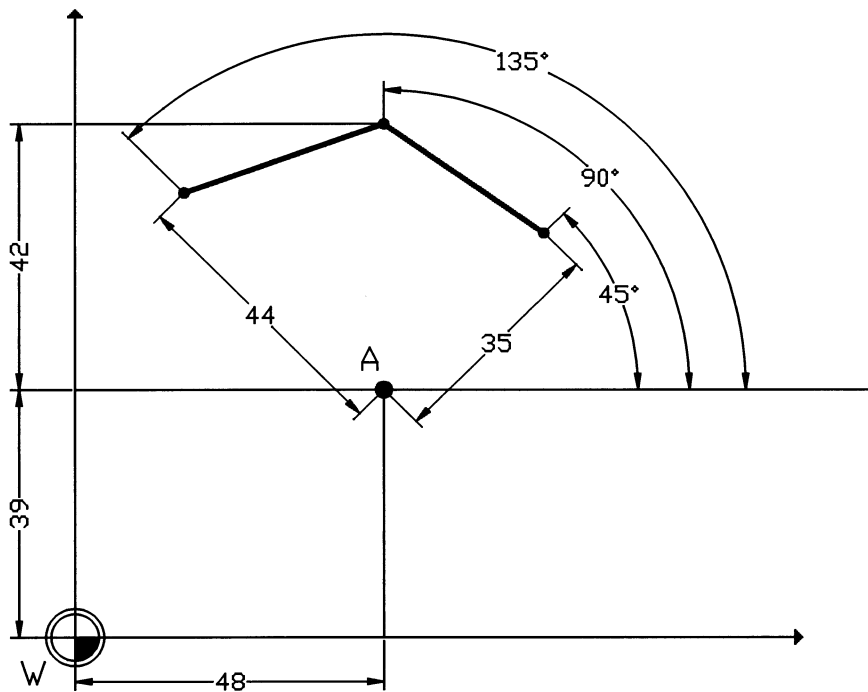
Polar programming with pole point of the centre point and end point is possible in G2 and G3 blocks.



ICP/Geometry calculation G64

G1, G2 and G3 blocks with B2=, B3= and L3= programming can be programmed in G64 and ICP. These blocks relate to the active pole point. The pole point can only be changed in G64 and not in ICP.

Example



B = new pole point

```
N30 G9 X48 Y39
N40 G1 B2=135 L2=44
N50 G1 B2=90 L2=42
N60 G1 B2=45 L2=35
```

Explanation:

N30 Definition of new pole point

N40 Definition of end point coordinates related to new pole point

10. One/Two point or two line geometry with chamfer or rounding G11

Note

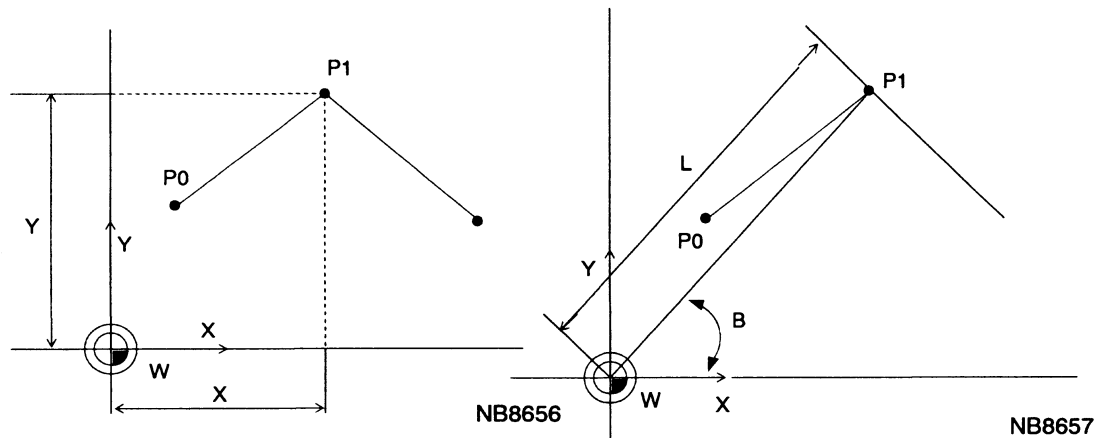
Use of this function is limited only to programs made on earlier control systems.
The operator can easily make programs requiring geometry calculations, using Interactive Contour Programming (ICP).

Purpose

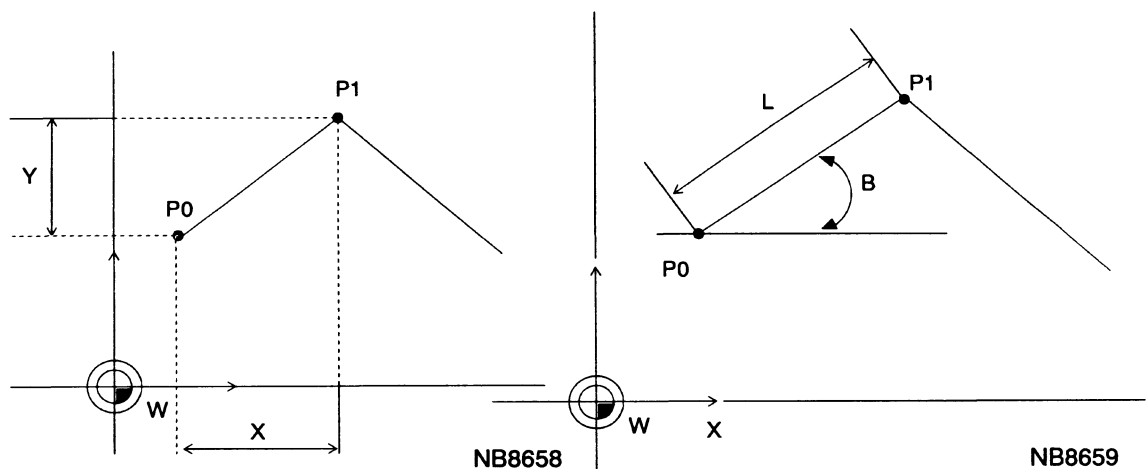
1. One Point Geometry

To program in one block:

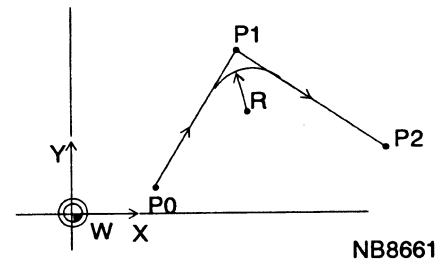
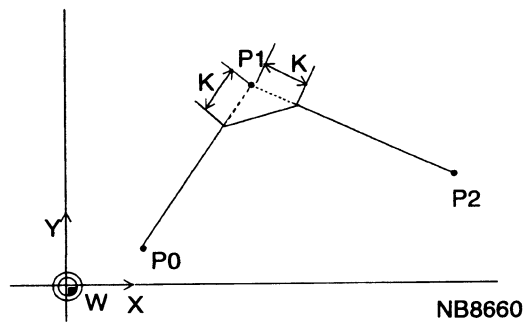
- the end point of a linear movement
- if required, a symmetrical chamfer or rounding between this movement and the next linear movement.



Absolute coordinates (G90)



Incremental coordinates (G91)

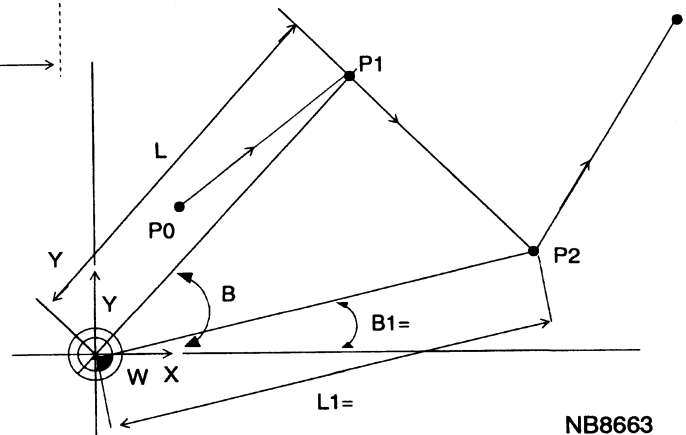
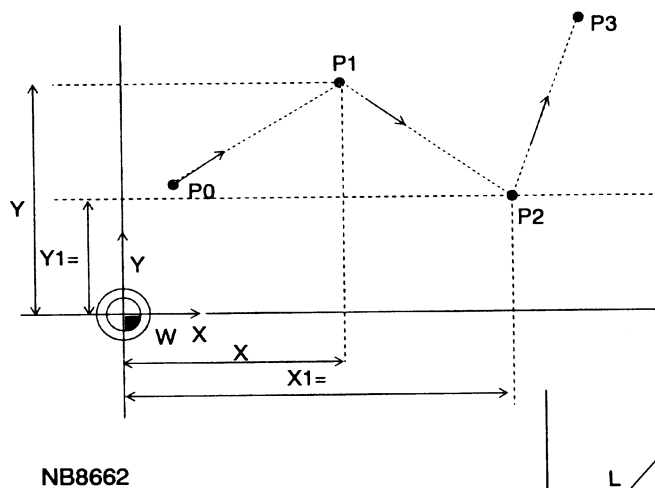


Chamfer rounding with one point geometry

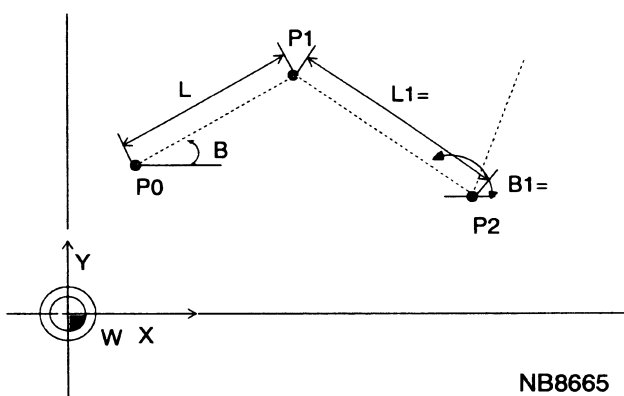
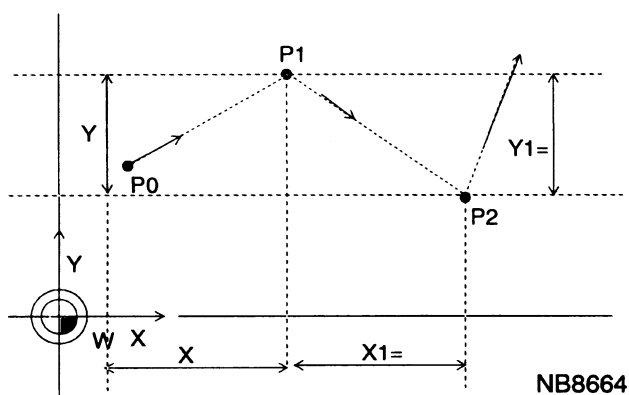
2. Two Point Geometry

To program in one block:

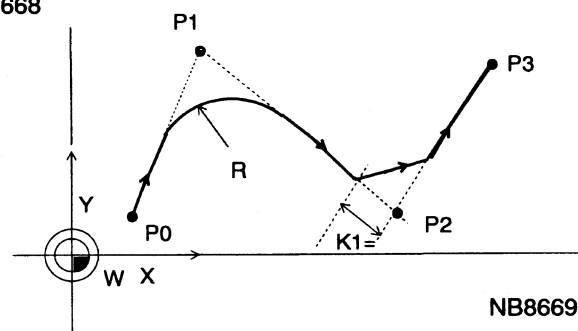
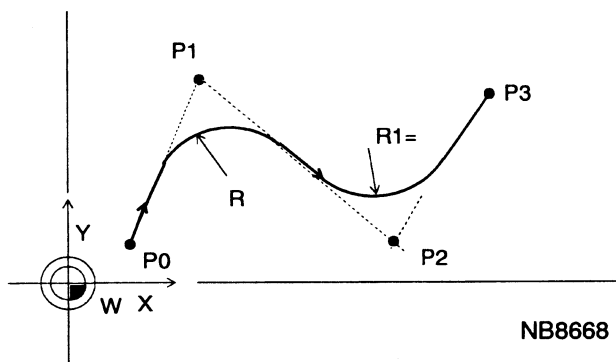
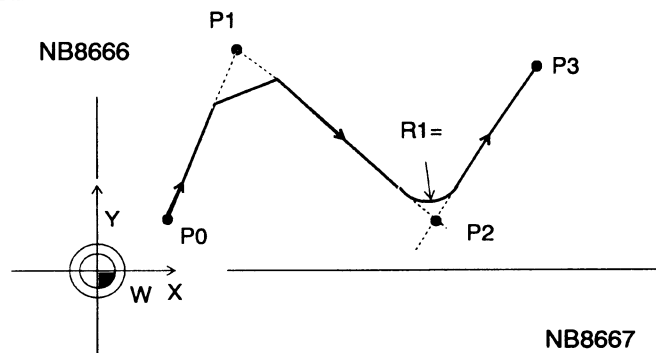
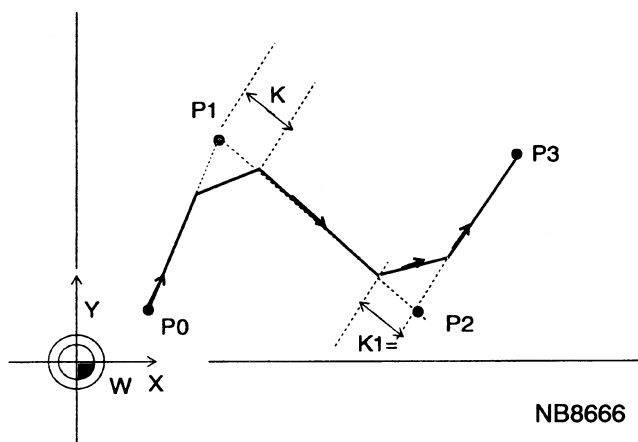
- the end points of two separate linear movements
- if required, a symmetrical chamfer or rounding between these movements
- if required, a symmetrical chamfer or rounding between the last movement and the next linear movement.



Absolute coordinates (G90)



Incremental coordinates (G91)

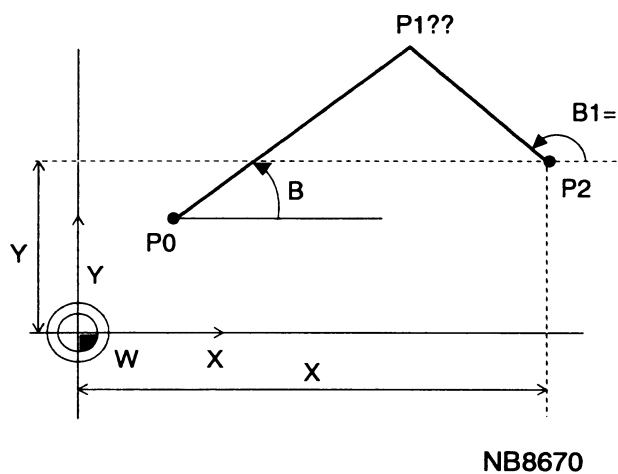


Chamfers or roundings with two point geometry

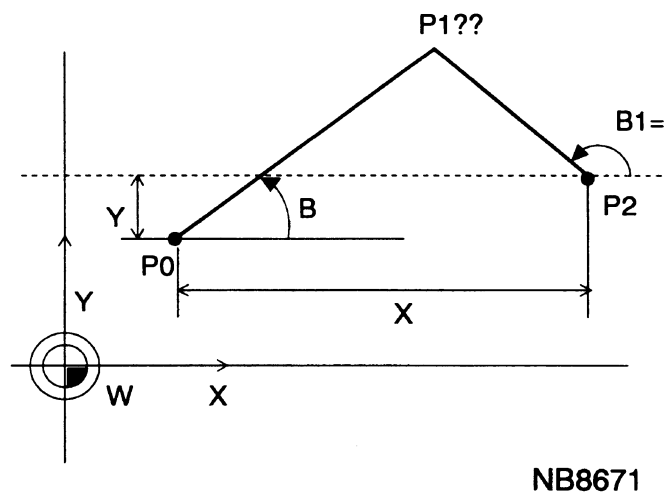
3. Two Line Geometry

To program in one block two separate linear movements:

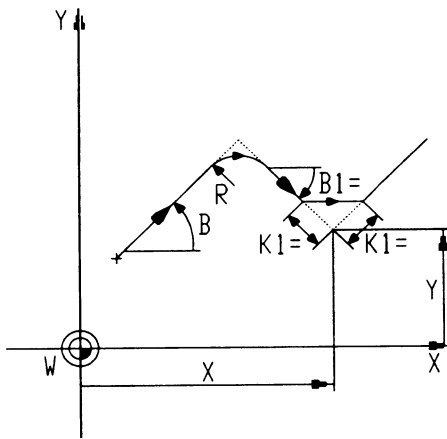
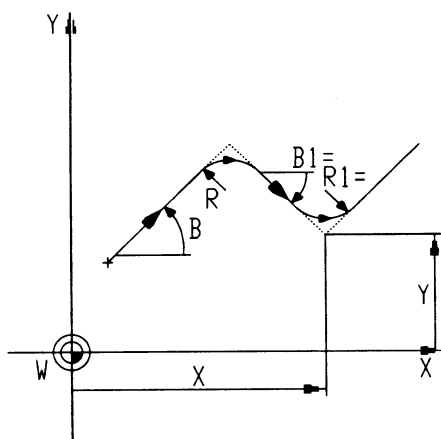
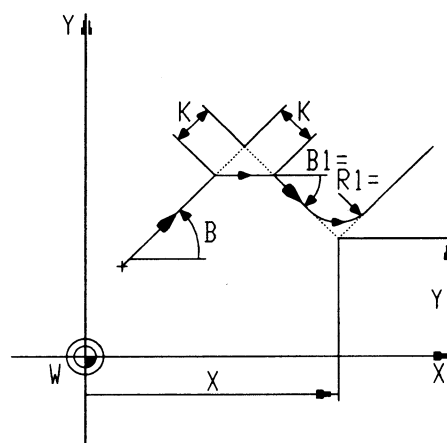
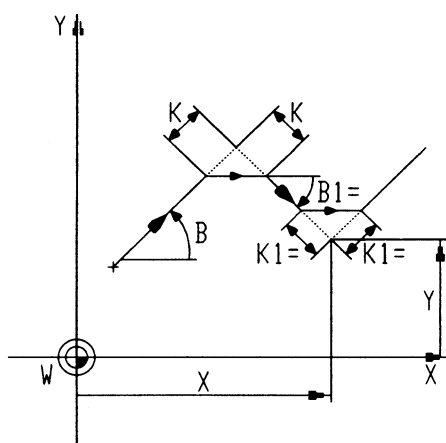
- the first linear movement with the angle with the main axis
- the second linear movement with the end point and the angle with the main axis
- if required, a symmetrical chamfer or rounding between these movements
- if required, a symmetrical chamfer or rounding between the last movement and the next linear movement.



Absolute coordinates (G90)



Incremental coordinates (G91)



Chamfers or roundings with two line geometry

Formats

One point geometry (XY-plane)

N... G11 X... Y... {Z...} {K...} {R...} {F...}

N... G11 B... L... {Z...} {K...} {R...} {F...}

Two point geometry (XY-plane)

N... G11 X... Y... X1=... Y1=... {K...} {R...} {K1=...} {R1=...} {F...}

N... G11 B... L... X1=... Y1=... {K...} {R...} {K1=...} {R1=...} {F...}

N... G11 X... Y... B1=... L1=... {K...} {R...} {K1=...} {R1=...} {F...}

N... G11 B... L... B1=... L1=... {K...} {R...} {K1=...} {R1=...} {F...}

Two line geometry (XY-plane)

N... G11 B... X... Y.. B1=... {K...} {R...} {K1=...} {R1=...} {F...}

Parameters

One point geometry

X,Y,Z Linear axis coordinates (absolute/incremental)

P Point definition number.

P1= Point definition number.

B	G90 active:	Angle the line through the datum point W and the end point makes with the X-axis (G17 and G18) or -Z-axis (G19)
	G91 active:	Angle the line makes with the X-axis (G17 and G18) or -Z-axis (G19)
L	G90 active:	length measured from the datum point W to the end point
	G91 active:	length measured from the last tool position to the end point

Two point geometry

X,Y,Z Linear axis coordinates of the first point (absolute/incremental). No tool axis allowed.

P1= Point definition number of the first point.

B	G90 active:	Angle the line through the datum point W and the first end point makes with the X-axis (G17 and G18) or -Z-axis (G19)
	G91 active:	Angle the first line makes with the X-axis (G17 and G18) or -Z-axis (G19)
L	G90 active:	length measured from the datum point W to the first end point
	G91 active:	length measured from the last tool position to the first end point

X1=,Y1=,Z1= Linear axis coordinates of the second point (absolute/incremental). No tool axis allowed.

P2= Second point definition number.

B1=	G90 active:	Angle the line through the datum point W and the second end point makes with the X-axis (G17 and G18) or -Z-axis (G19)
	G91 active:	Angle the second line makes with the X-axis (G17 and G18) or -Z-axis (G19)
L1=	G90 active:	length measured from the datum point W to the second end point
	G91 active:	length measured from the first end point to the second end point

Two line geometry

X,Y,Z Linear axis end point coordinates of the second line (absolute/incremental). No tool axis allowed.

P Point definition number of the end point of second line.

P1= Point definition number of the end point of second line.

B Angle the first line makes with the X-axis (G17 and G18) or -Z-axis (G19)

B1= Angle the second line makes with the X-axis (G17 and G18) or -Z-axis (G19)

Words for chamfer or rounding in the three cases

K First chamfer length

R First rounding radius

K1= Second chamfer length

R1= Second rounding radius

Modal words

F,S,T,H,M,E...=

Ti=, T2=

Associated functions

G1

Type of function

Non-modal

Notes and usage

FEEDRATE

All movements in a G11-block use the last programmed feedrate if a new feedrate is not stated in the G11-block.

NEXT MOVEMENT AFTER A G11 BLOCK

If a second chamfer (K1=) or a second rounding (R1=) has been programmed, the block following the G11-block must contain either a G1 or G11-function.

If a G1-block has been programmed following a G11-block, both end point coordinates (e.g. X.. and Y..) must be stated.

TOOL AXIS PROGRAMMING WITH A G11

With the one point geometry it is allowed to program the tool axis with the end point coordinates. The three axis movement is executed before making the chamfer or rounding programmed in the same G11-block.

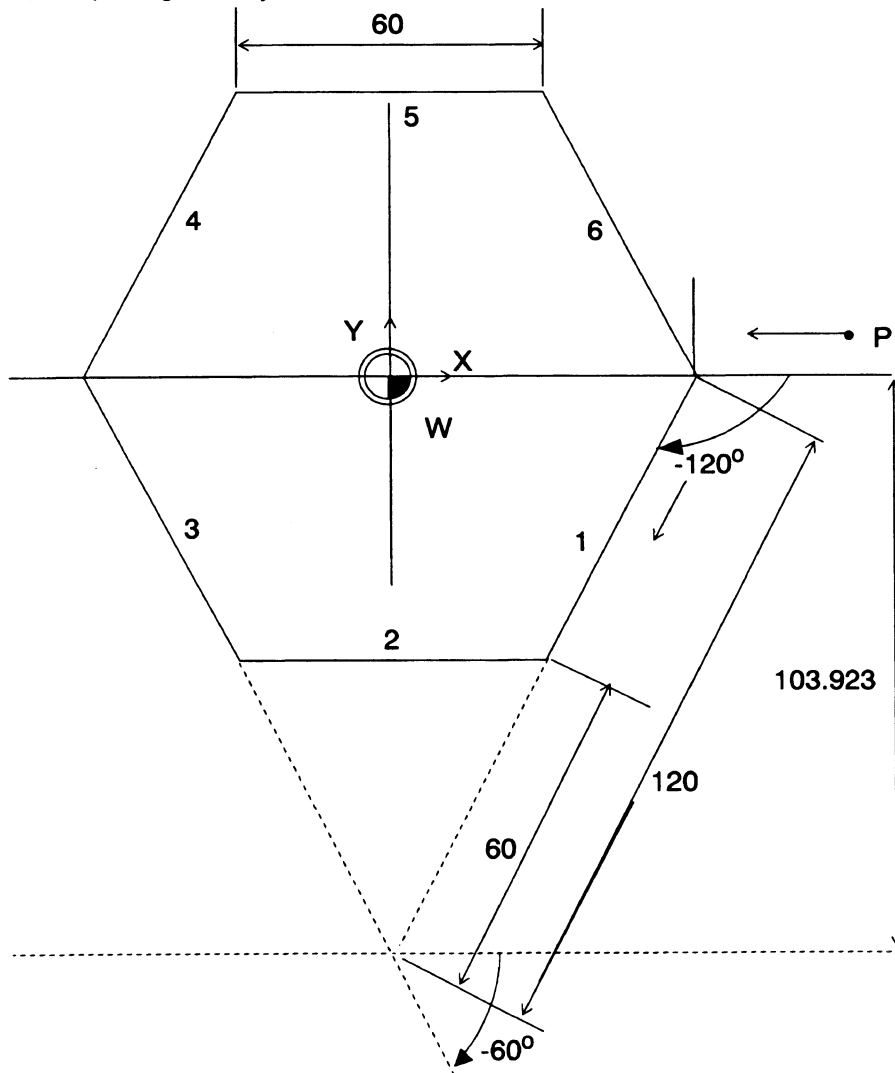
With the two point or two line geometry tool axis programming is not allowed.

RESTRICTION

1. The G11 function is not allowed once the geometry is activated (G64 active).
2. The G11 function is not allowed for defining a pocket or island contour. (G200..G208)
3. G11 is not allowed with a programmed tool axis. In case programs are started with above mentioned programming combination, the programmer may encounter operational errors PO1 and/or P34 at execution of the program.

Examples

EXAMPLE 1. One point geometry



NB8579

The regular hexagon has to be milled on the outside of the workpiece surface. The one point geometry with angle is used. The sides 2 and 4 are programmed as chamfers.

N9010
N1 G17 T1 M6
N2 G0 X100 Y10 Z-10 S1000 M3
N3 G1 F300
N4 G43 X60
N5 G41 Y0
N6 G11 B-90 L103.923 K60
N7 G11 B150 L103.923 K60
N8 G11 B60 L60
N9 G11 B0 L60
N10 G40
N11 G1 X100 Y10
N12 G0 Z100 M30

Explanation

N9010: Program number.

N1: Activate the main plane. Load the tool

N2: Start the spindle, move tool to point P and to depth.

N3: Set feedrate to 300 mm/min.

N4: Move the tool to the corner of the hexagon.

N5: Set radius compensation LEFT.

N6: Mill along sides 1 and 2. Programmed is the intersection point of sides 1 and 3 the chamfer (K-word) around this point.

N7: Mill along sides 3 and 4. Programmed is:
- the intersection point of sides 3 and 5
- the chamfer (K-word) around this point.

N8: Mill along side 5.

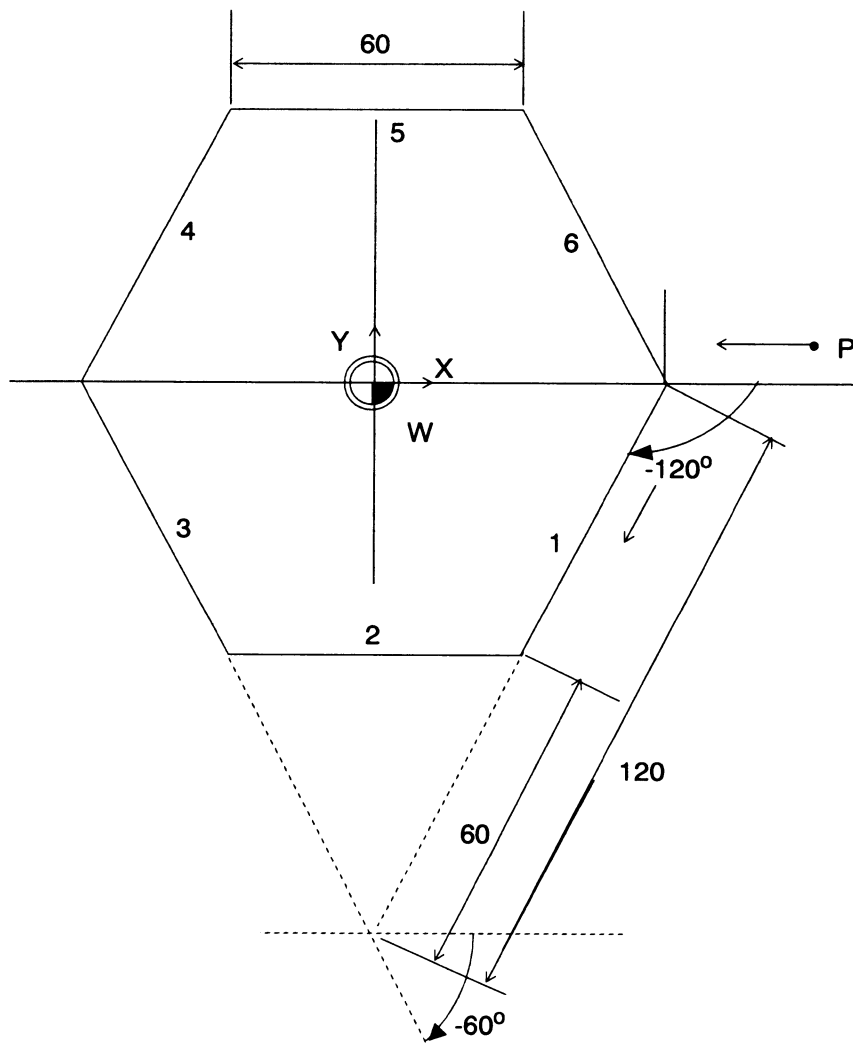
N9: Mill along side 6.

N10: Cancel the radius compensation.

N11: Move tool away from part.

N12: Retract the tool and end of program.

EXAMPLE 2. Two point geometry



NB8580

The regular hexagon has to be milled on the outside of the workpiece's surface. The two point geometry with angles and increments is used. The sides 2 and 5 are programmed as chamfers.

```

N9011
N1 G17 T1 M6
N2 G0 X100 Y10 Z-10 S1000 M3
N3 G1 F300
N4 G43 X60
N5 G41 Y0
N6 G91
N7 G11 B-120 L120 K60 B1=120 L1=120
N8 G11 B60 L120 K60 B1=-60 L1=120
N9 G40
N10 G90
N11 G1 X100 Y10
N12 Z10 M30

```

Explanation:

N9011: Program number.

N1: Activate the main plane. Load tool 1

N2: Start the spindle, move tool to point P and then to depth.

N3: Enter the linear movement and set the feedrate.

N4: Move the tool to the corner of the hexagon.

N5: Set radius compensation LEFT.

N6: Activate the incremental mode. The length values in the next blocks are measured from the previous tool position.

N7: Mill along sides 1, 2 and 3. Programmed is

- the intersection point of sides 1 and 3 (B and L),
- the chamfer (K-word) around this point
- the end point of side 3 (B1= and L1=).

N8: Mill along sides 4, 5 and 6. Programmed is

- the intersection point of sides 4 and 6 (B and L),
- the chamfer (K-word) around this point
- the end point of side 6 (B1= and L1=)

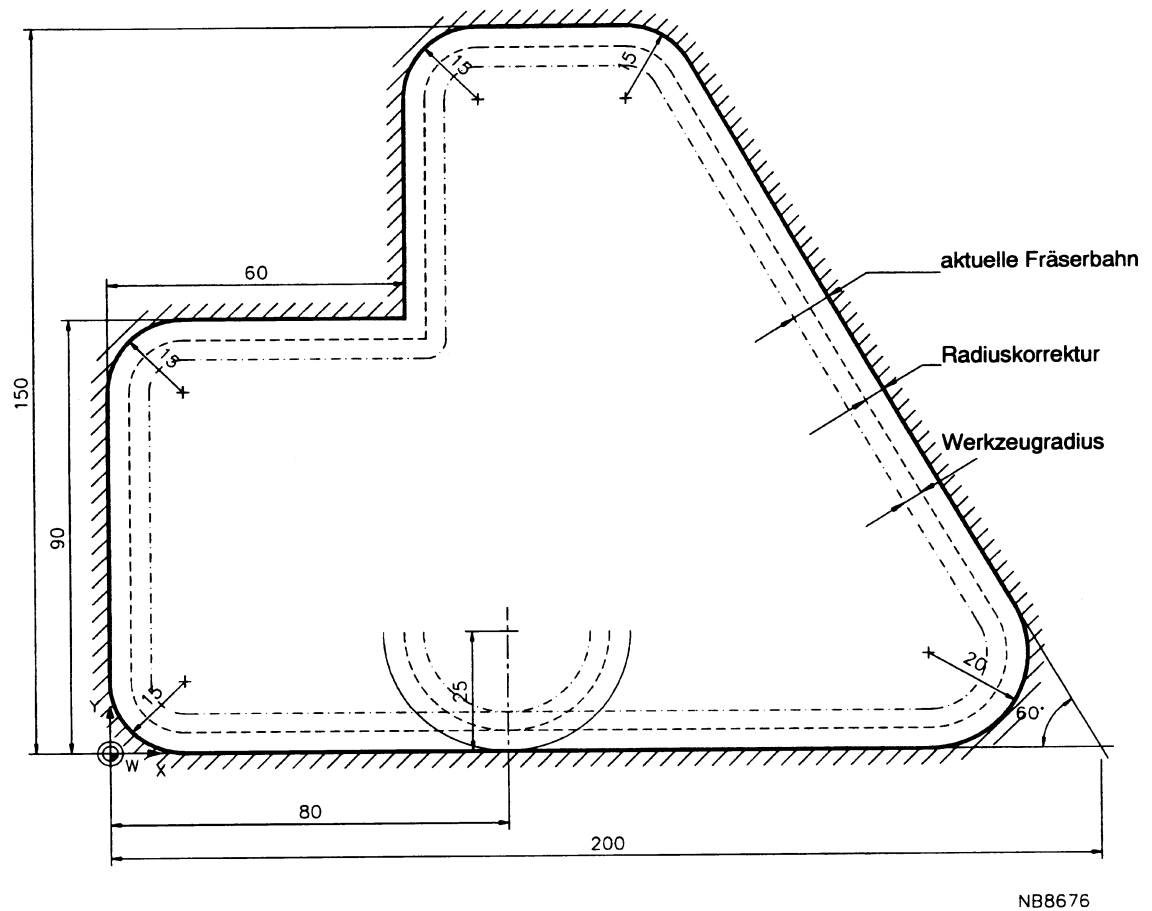
N9: Cancel radius compensation.

N10: Set the absolute mode.

N11: Move tool away from workpiece.

N12: End of program.

EXAMPLE 3. Two line geometry



The inside pocket can be programmed using G11-function with two line geometry elements.

```

N9012
N1 G17
N2 X80 Y25 Z0 T1 M6
N3 G1 Z-10 F75 S1000 M3
N4 G43 X105 F100
N5 G42
N6 G2 X80 Y0 R25
N7 G11 X0 Y90 C0 B1=90 R15 R1=15
N8 G11 X60 Y150 C0 B1=90 R1=15
N9 G11 X200 Y0 C0 B1=120 R15 R1=20
N10 G1 X80 Y0
N11 G2 X55 Y25 R25
N12 G40
N13 G0 Z200 M30
    
```

Explanation:

N9012: Program number.
 N1: Activate XY-plane (G17)
 N2: Load tool T1 (Mill diameter 10mm). Move the tool to point B and above the workpiece.
 N3: Start the spindle and feed to depth.
 N4: Move the tool to the starting point of the entering-circle.
 N5: Set radius compensation RIGHT.
 N6: Move to the contour via the entering-circle.
 N7: Mill - along the X-axis, (B0)
 - along the radius, (R15)
 - along the Y-axis, (B1 =90)
 - along the second radius (R1 =15).
 N8: Mill - parallel to the X-axis, (B0)
 - parallel to the Y-axis, (B1 =90)
 - along the second radius (R1 =15)
 N9: Mill - parallel to the X-axis, (B0)
 - follow the first radius, (R15)
 - mill along the slope of 60 degrees, (B1=120)
 - follow the second radius (R20).
 N10: Mill along the X-axis to the starting point of the circle for leaving the contour.
 N11: Exit the contour with a circular movement.
 N12: Cancel the radius compensation.
 N13: Retract the tool. End of program.

11. Repeat function G14

Purpose

To repeat the execution of a specified number of blocks within a partprogram or subprogram.

Format

N... G14 N1=... {N2=...} {J...} {K...}

Parameters

N1= Repeater begin block
 N2= Repeater end block
 J Number of repeats
 K Repeat decrement
 ENNN= Parameter definition

Modal parameters
 E...= E-parameter definition.

Associated functions

G22, G29

Type of function

Non-modal

Notes and usage

BLOCK NUMBERS OF REPEAT SEQUENCE (N1=, N2=)

These block numbers must be in the same partprogram or subprogram.

If N2= is not programmed, only the block indicated by N1 = is repeated the specified number of times.

ORDER OF BLOCKS TO BE REPEATED

The order of executing the blocks in the repeat sequence must be the same as the order originally programmed. So in the program block N1=.. must be before block N2=..

NUMBER OF REPEATS (J)

The number of repeats is programmed with the J-word. The J-word is not necessarily an integer value. The integer part, thus the part before the decimal point, is used as the number of repeats.

When no number of repeats is programmed (no J-word is present), the sequence is repeated only once.

REPEAT DECREMENT (K)

The K-word allows the value of the J-word to be recalculated and used as the condition for repeating.

If the K-word is not programmed, the value of the J-word is reduced by 1 after every repeat.

If $K > 0$, the value is used to reduce the value of the J-word. If eg. K5 were programmed, 5 would be subtracted from the value of the J-word after every repeat. As long as the J-word is greater than 0, a repeat is executed.

If $K \leq 0$ is programmed, an error message is displayed.

NESTING OF REPEATS

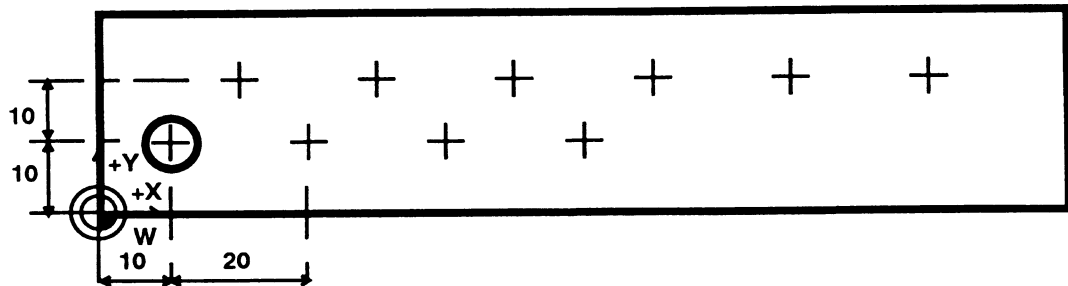
A repeating block sequence may be included in another repeating block sequence; this can be done four times.

CONTINUATION AFTER THE REPEAT

Once the repeats are executed, the program continues with the block after the G14.

Example

EXAMPLE 1. Programming a repeat function



N1234
N1 G195 X-10 Y-10 Z10 I160 J50 K-30
N2 G99 X0 Y0 Z0 I140 J30 K-10
N3 G17
N4 T1 M6
N5 G81 Y5 Z-11.5 F0.2 S2000 M3
N6 G79 X10 Y10 Z0
N7 G79 L1=20 B1=0
N8 G14 N1=5 J2
N9 G92 X10 Y10
N10 G14 N1=4 N2=6
N11 G14 N1=7 N2=7 J2
N12 G93 X0 Y0
N13 G0 Z200
N14 M30

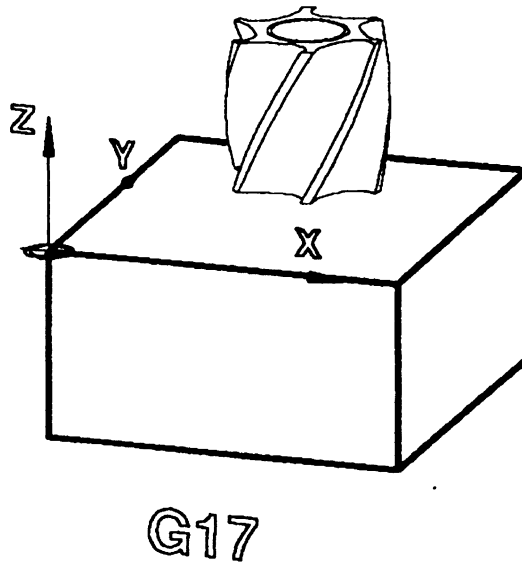
Explanation:

- | | |
|------|---|
| N1: | Set up graphic window |
| N2: | Set up graphic (material) |
| N3: | Define the main plane. |
| N4: | Load tool 1 with a drill diameter of 10mm. |
| N5: | Define fixed drilling cycle and start the spindle |
| N6: | Drill holes |
| N7: | Drill holes with Polar coordinates. |
| N8: | Programblock N7..N8 will be repeated twice. |
| N9: | Absolute zero point shift |
| N10: | Programblock N7..N8 will be repeated once. |
| N11: | Programblock N7 will be repeated twice. |
| N12: | Absolute zero point shift. |
| N13: | Retract the tool to Z200. |
| N14: | End of program. |

12. Mainplane XY, tool Z G17

Purpose

The main spindle of the machine tool determines the position of the tool axis. With G17 is defined that the tool axis is the Z-axis and the main plane for milling operations the XY-plane.



Format

N... G17

Parameters

Modal words
H, T1=

Associated Functions

G18/G19

Type of function

Modal

Notes and usage

DEFAULT PLANE

When switching on the machine or after a CLEAR CONTROL the machine activates automatically due to MC11 (0=G17, 1=G18, 2=G19) a PLANE.

The last selected PLANE is active when the machine is switch on normally.

OPERATIONS IN THE PLANE

Calculations for radius compensation, the geometry (G64), polar coordinates, milling cycles, the pocket cycle, etc. are performed in the current plane. Thus when G17 is active the XY-plane.

OPERATIONS IN THE TOOL AXIS

Tool length compensation and the fixed cycles for hole operations use the current tool axis. Thus when G17 is active the Z-axis.

ANGULAR HEAD

When an angular head is fitted, the axis configuration of the machine tool remains unchanged. So the tool can be in either the Y- or X-axis.

With the function G18 or G19 is programmed in which axis the tool is standing and which plane is the plane of operation.

G18: XZ-plane, tool in Y-axis

G19: YZ-plane, tool in X-axis

Refer to G19 using an angular head for programming an angular head.

CHANGING THE PLANE OF OPERATION

When a new plane is selected, thus either G18 or G19 activated, the length compensation in the Z-axis is cancelled and activated in the tool axis related to the selected plane.

CANCELLATION

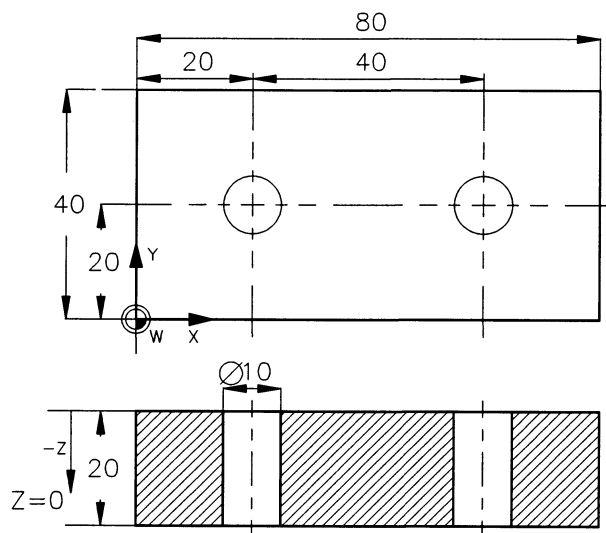
The G17 function is cancelled by activating another machining plane, using either the G18 or G19 functions. The G17 function is not cancelled by CLEAR CONTROL or by softkey CANCEL PROGRAM.

TOOL OFFSETS

Tool dimensions stored in the Tool Memory are independent of the selected plane.

Example

EXAMPLE 1: G17



NB8589

```

N9001
N1 G17
N2 T1 M6
N3 G0 X20 Y20 Z1 F400 S1600 M3
N4 G1 Z-23.5
N5 G0 X60 Z1
N6 G1 Z-23.5
N7 G0 Z200 M30

```

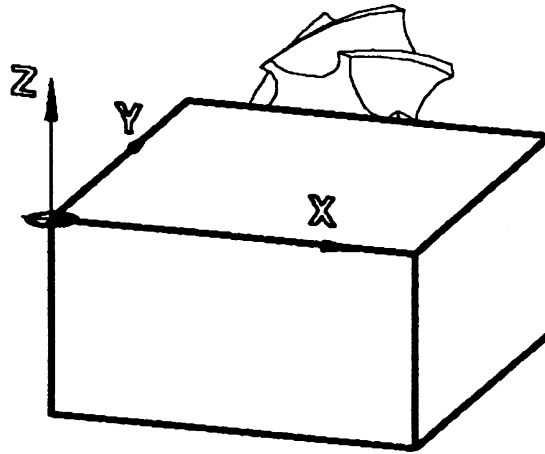
Explanation:

- N1: Activate G17 main plane.
- N2: Load tool T1 and its offsets. Drill diameter is 10 mm.
- N3: Move tool rapidly (G0) to programmed position. Set feedrate to 400 mm/min. Make spindle rotate clockwise (M3) at 1600 rev/min.
- N4: Feed tool to programmed depth.
- N5: Retract tool to Z1 and then move the tool rapidly to X60. The CNC's positioning logic ensues that the tool does not collide with the workpiece, because the tool is first moved along the Z-axis before moving along the X-axis.
- N6: Feed tool to programmed depth.
- N7: Retract tool to Z200 and end of program.

13. Main plane XZ, tool Y G18

Purpose

The main spindle of the machine tool determines the position of the tool axis. With G18 is defined that the tool axis is the Y-axis and the main plane for milling operations the XZ-plane.



G18

Format

N... G18

Parameters

Modal words
H, T1=

Associated Functions

G17/G19

Type of function

Modal

Notes and usage

DEFAULT PLANE

When switching on the machine or after a CLEAR CONTROL the machine activates automatically due to MC11 (0=G17, 1=G18, 2=G19) a PLANE.

The last selected PLANE is active when the machine is switch on normally.

OPERATIONS IN THE PLANE

Calculations for radius compensation, the geometry (G64), polar coordinates, milling cycles, the pocket cycle, etc. are performed in the current plane. Thus when G18 is active the XZ-plane.

OPERATIONS IN THE TOOL AXIS

Tool length compensation and the fixed cycles for hole operations use the current tool axis. Thus when G18 is active the Y-axis.

ANGULAR HEAD

When an angular head is fitted, the axis configuration of the machine tool remains unchanged. So the tool can be in either the Z- or X-axis.

With the function G17 or G19 is programmed in which axis the tool is standing and which plane is the plane of operation.

G17: XY-plane, tool in Z-axis

G19: YZ-plane, tool in X-axis

Refer to G19 using an angular head for programming an angular head.

CHANGING THE PLANE OF OPERATION

When a new plane is selected, thus either G17 or G19 activated, the length compensation in the Y-axis is cancelled and activated in the tool axis related to the selected plane.

CANCELLATION

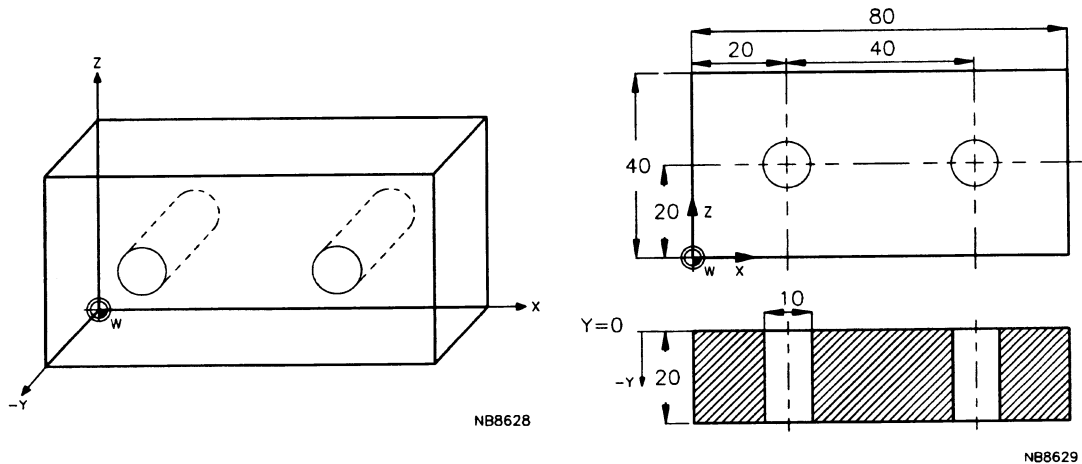
The G18 function is cancelled by activating another machining plane, using either the G17 or G19 functions. The G18 function is not cancelled by CLEAR CONTROL or by softkey CANCEL PROGRAM.

TOOL OFFSETS

Tool dimensions stored in the Tool Memory are independent of the selected plane.

Example

EXAMPLE 1: A vertical milling machine is assumed.



```

N9002
N1 G18
N2 T2 M6
N3 G0 X20 Y1 Z20 F400 S1000 M3
N4 G1 Y-23.5
N5 G0 X60 Y1
N6 G1 Y-23.5
N7 G0 Y200 M30

```

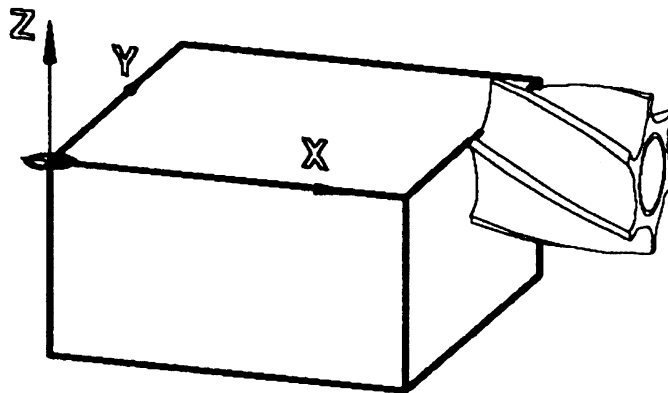
Explanation:

- N1: Make XZ-plane (G18) active.
- N2: Load tool T2 and its offsets. Drill diameter is 10 mm.
- N3: Move tool rapidly (G0) to the programmed position. Set the feedrate to 400 mm/min and make spindle rotate clockwise (M3) at 1000 rev/min.
- N4: Feed tool to depth.
- N5: Retract tool to Y1 and then move tool rapidly to X60. The CNC's positioning logic ensures that the tool does not collide with the workpiece, because the tool is first moved along the Y-axis before moving along the X-axis.
- N6: Feed tool to depth.
- N7: Retract tool to Y200 and end of program.

14. Mainplane YZ, tool X G19

Purpose

The main spindle of the machine tool determines the position of the tool axis. With G19 is defined that the tool axis is the X-axis and the main plane for milling operations the YZ-plane.



G19

Format

N... G19

Parameters

Modal words
H, T1=

Associated Functions

G17/G18

Type of function

Modal

Notes and usage

DEFAULT PLANE

When switching on the machine or after a CLEAR CONTROL the machine activates automatically due to MC11 (0=G17, 1=G18, 2=G19) a PLANE.

The last selected PLANE is active when the machine is switch on normally.

OPERATIONS IN THE PLANE

Calculations for radius compensation, the geometry (G64), polar coordinates, milling cycles, the pocket cycle, etc. are performed in the current plane. Thus when G19 is active the YZ-plane.

OPERATIONS IN THE TOOL AXIS

Tool length compensation and the fixed cycles for hole operations use the current tool axis. Thus when G19 is active the X-axis.

ANGULAR HEAD

When an angular head is fitted, the axis configuration of the machine tool remains unchanged. So the tool can be in either the Z- or Y-axis.

With the function G17 or G18 is programmed in which axis the tool is standing and which plane is the plane of operation.

G17: XY-plane, tool in Z-axis

G18: XZ-plane, tool in Y-axis

Refer to using an angular head for programming an angular head.

CHANGING THE PLANE OF OPERATION

When a new plane is selected, thus either G17 or G18 activated, the length compensation in the X-axis is cancelled and activated in the tool axis related to the selected plane.

CANCELLATION

The G19 function is cancelled by activating another machining plane, using either the G17 or G18 functions. The G19 function is not cancelled by CLEAR CONTROL or by softkey CANCEL PROGRAM.

TOOL OFFSETS

Tool dimensions stored in the Tool Memory are independent of the selected plane.

USING AN ANGULAR HEAD

When an angular head is used, its dimensions must be programmed. Either a zero point shift (G92 or G93) or a stored zero offset (G54 - G59) can be used for this purpose. The use of stored zero offsets is recommended because the part program remains independent of the dimensions of the angular head.

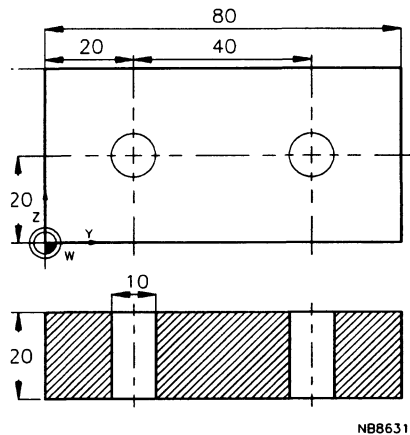
TOOL IN + OR - DIRECTION OF X-AXIS

Especially with an angular head in the X-axis the tool can be in the positive (+) or negative (-) direction of the axis. The functions G66 and G67 are available to indicate in which direction the tool is standing and allows the part programmer to look always in the same way at the plane of operation.

Refer to G66/G67 for using these functions.

Example

EXAMPLE 1: A vertical milling machine is assumed.



```

N9003
N1 G19
N2 T3 M6
N3 G0 X1 Y20 Z20 F400 S1000 M3
N4 G1 X-23.5
N5 G0 X1 Y60
N6 G1 X-23.5
N7 G0 X200 M30

```

Explanation:

- N1: Make YZ-plane (G19) active.
- N2: Select tool T3 and its offsets. Drill diameter is 10 mm.
- N3: Move tool rapidly (G0) to programmed position. Set the feedrate to 400 mm/min and make spindle rotate clockwise (M3) at 1000 rev/min.
- N4: Feed tool to depth.
- N5: Retract tool to X1 and then move tool rapidly to Y60. The CNC's positioning logic ensures that the tool does not collide with the workpiece, because the tool is first moved along the X-axis before moving along the Y-axis.
- N6: Feed tool to depth.
- N7: Retract tool to X200 and end of program.

15. Macro call G22

Purpose

To execute a subprogram with standard operations.

Format

To call a subprogram.

N... G22 N=... {E...=}

To activate a subprogram on the condition that E...>0

N... G22 E... N=... {E...=}

Parameters

E Parameter definition

N= Macro number

Modal Words

ENNN= Parameter definition

Associated functions

G14, G29

Type of function

Non-modal.

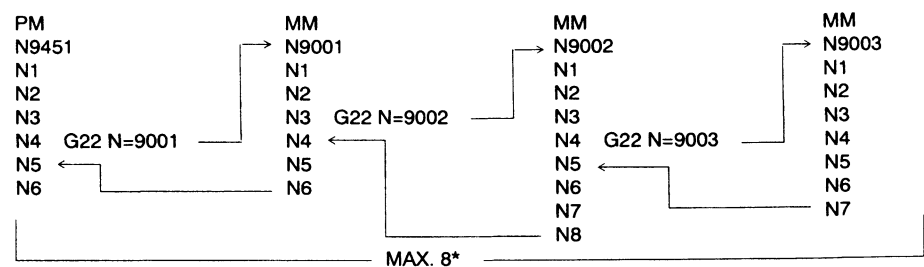
Notes and usage

ACTIVATION

A subprogram is completely executed when it is called from a main program or from another subprogram.

NESTING OF SUBPROGRAMS

When a subprogram calls another subprogram, the subprogram which is activated is referred to as a 'nested' subprogram. At the end of a nested subprogram the calling subprogram continues. A maximum of eight 'nested' programs can be used.



E-PARAMETERS

A subprogram may contain E-parameters which are variables whose values are stored in a separate CNC memory. Subprograms can therefore be written which have a general application. When the dimensions of a component are known, only the E-parameter values need to be altered, not the program.

E-parameters can get their value in the main program or subprogram, via the operator's panel, or by reading-in the parameter memory.

Arithmetical calculations with parametric values are allowed in programs and subprograms. The same parameter can be used by different (sub) programs.

Refer to the special appendix about E-parameters at the end of this manual for more details of programming with E-parameters.

NUMBER OF PARAMETER DEFINITIONS

In a block with a macro call up to 10 parameters can get their value. If more parameters are used, extra lines before the macro call are necessary.

EVALUATION OF DEFINED PARAMETERS

Any value or arithmetical expression can be assigned to a parameter in a G22-block.

Parameters programmed in the G22 block are evaluated and calculated before the execution of the macro.

CONTINUATION AFTER THE MACRO CALL

Once the macro is executed, the program continues with the block after the G22 in which the macro was called.

CONDITIONAL MACRO CALL (E)

The value of the E-word is used for dictating if a conditional macro call must be performed.

If the value of $E... > 0$, the macro call is performed.

After the call the program continues with the block after the G22. Parameter $E...$ is not influenced by the macro call.

If the value of $E... \leq 0$, the macro is not called. The program continues with the block after the G22.

Examples

EXAMPLE 1. A macro call.

N100 G22 N=9100 E1=24 E2=3

Explanation:

N150: Execute subprogram N9100 with parameters E1=24 and E2=3.

EXAMPLE 2. Conditional macro call.

N150 G22 E60 N=9100

Explanation:

N150: Execute subprogram N9100 when the value of E60 > 0.

EXAMPLE 3. Macro without E-parameters.

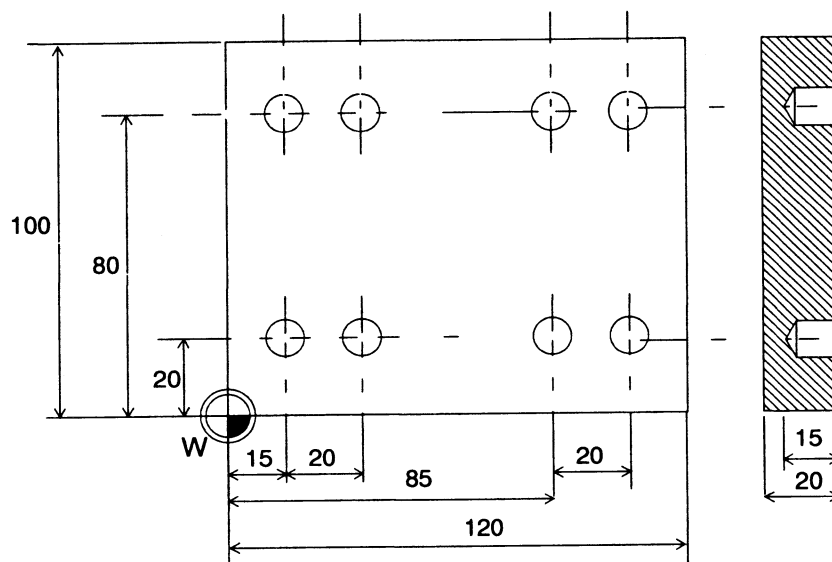
Subprogram for drilling two holes:

```
N9001
N1 G91
N2 G1 Z-16 M8
N3 G0 Z16 M9
N4 X20
N5 G1 Z-16 M8
N6 G0 Z16 M9
N7 G90
```

Explanation of the macro:

```
N1:  Activate incremental programming.
N2:  Switch coolant ON. Move tool with a feedrate in negative direction.
N3:  Retract tool. Switch coolant OFF.
N4:  Move tool 20 mm along X-axis to second start position.
N5:  Switch coolant ON. Feed tool 15 mm into workpiece.
N6:  Retract tool. Switch coolant OFF.
N7:  Re-activate absolute programming.
```

EXAMPLE 4: Main program for drilling four pairs of holes:



NB8537

Main program	Macro program
N45 T1 M6	N9001
N50 F400 S1600 M3	N1 G91
N55 X15 Y20 Z1	N2 G1 Z-15
N60 G22 N=9001	N3 G0 Z16
N65 X85	N4 G90
N70 G22 N=9001	
N75 X85 Y80	
N80 G22 N=9001	
N85 X15	
N90 G22 N=9001	

Explanation of the main program:

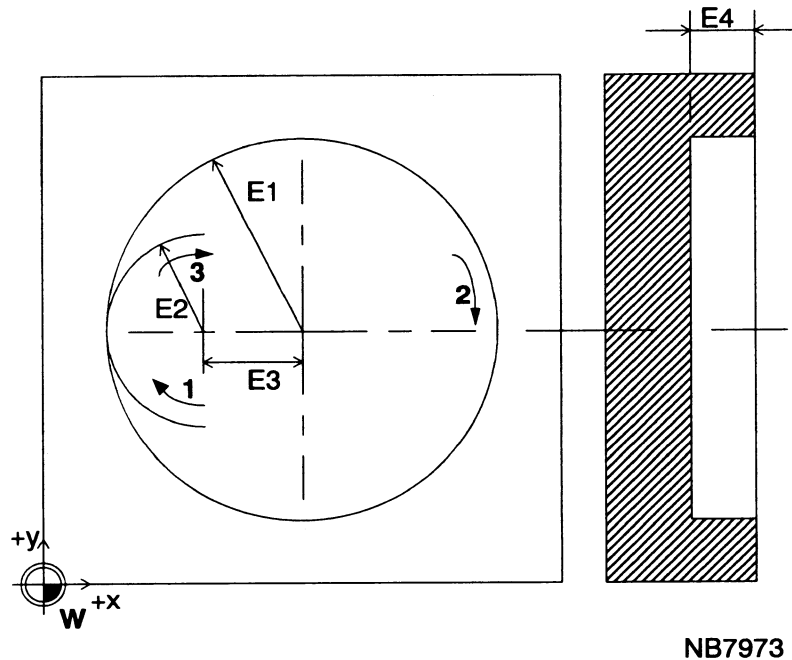
N45: Load tool T1 and use its offsets. Drill diameter is 10 mm.
 N50: Make spindle rotate clockwise at 1600 rev/min. Feedrate at 400 mm/min
 N55: Move tool to first drilling position and 1 mm off top surface.
 N60: Activate subprogram
 N65: Move tool to second drilling position.
 N70: Activate subprogram
 N75: Move tool to third drilling position.
 N80: Activate subprogram
 N85: Move tool to fourth drilling position.
 N90: Activate subprogram

Programming instruction

G0 is not really necessary in N65, N75 and N85 blocks, because G0 is programmed in the last block of the subprogram. G0 has been programmed in the latter block, because then you need not know how exactly the subprogram ends in order to understand the main program.

EXAMPLE 5. Subprogram which uses E-parameters.

This example describes a general subprogram for finishing a milled full circle. To achieve a smooth transition the circle is entered and exited with a small circular arc. Tool radius compensation is used so that the workpiece contour can be programmed directly.



The following parameters must be set in the program:

- E1: Radius R1 of the full circle
- E2: Radius R2 of the approach and retraction circle
- E4: The depth, including a clearance distance.

The following parameter is calculated:

- E3: The distance $(R1-R2)$.

Programming instruction

Make the subprogram as universal as possible, i.e. without tool change, feed and speed. You will then also be able to use the subprogram for other tools. Set the tool change, feed and speed in the main program.

The subprogram is written with incremental dimensions.

```

N9002
N1 G91
N2 E3=E1-E2
N3 G1 Z=-E4
N4 X=-E3
N5 G43 Y=-E2
N6 G42
N7 G2 X=-E2 Y=E2 R=E2
N8 I=E1 J=0
N9 X=E2 Y=E2 R=E2
N10 G40
N11 G0 X=E3 Y=-E2 Z=E4
N12 G90

```

Explanation:

N1: Incremental programming is set.
 N2: Parameter E3 is calculated.
 N3: Feed tool to depth. In E4 a clearance from the workpiece must be considered.
 N4: Move tool to the centre of the approach circle.
 N5: Feed tool to the starting point of the approach circle.
 N6: Set radius compensation RIGHT.
 N7: Move tool in a clockwise direction to produce an arc. Both the end point and the radius are programmed with parameters.
 N8: Mill the complete circle. The circle centre is programmed.
 N9: Mill the arc of the retraction circle. Both the end point and the radius are programmed with parameters.
 N10: Cancel the radius compensation.
 N11: Move tool back to the centre of the circle and retract at rapid traverse rate.
 N12: Re-activate absolute programming (G90).

When this subprogram is used the calling program must contain:

- A tool movement to the circle centre (N200)
- A definition of the parameters (N201)
- A call of the subprogram (N201)

The calling program is as follows:

```

N9002
N1 G17
N2 G54
N3 T1 M6
N4 F900 S2000 M3
:
N200 G0 X75 Y80 Z0
N201 G22 N=9002 E1=30 E2=15 E4=15
:
N500 M30

```

After calling the subprogram, a circle with a diameter of 60 mm is milled in the required (X75, Y80)

16. Program call G23

Purpose

To call a partprogram from a main program.

Format

N... G23 N=... {E...=...}

Parameters

N= Program number

Modal Words

ENNN= Parameter definition

Associated functions

G14, G22, G29

Type of function

Non-modal.

Notes and usage

ACTIVATION

A called partprogram is completely executed when it is called from a main program.

PROGRAM SIZE

Programs smaller than 100 Kbyte will be stored in the work-memory and executed as a normal G23 call.

Programs bigger than 100 Kbyte cannot be stored in the work-memory. They will be separated automatically and invisible, in a lot smaller partprograms. These partprograms will be executed automatically (CAD MODE).

CONDITIONAL JUMPS

In the main program conditional jumps can be used to decide which program should be executed.

RESTRICTIONS

A called partprogram cannot contain a G23-function; part programs cannot be "nested" within one another.

A subprogram (macro) must not contain the G23-function.

Programs bigger than 100 Kbyte may not have jump instructions.

CONTINUATION AFTER THE PROGRAM CALL

Once the called program is executed, the main program continues with the block after the G23 in which the program was called.

INTERRUPTING A CALLED PROGRAM

If the execution of a called partprogram is terminated via CLEAR CONTROL, program control is back at the begin of the main program.

Example

```
N9990
:
N10 G23 N=988      Activate macro program
                   N988
                   N1
                   N2
                   :
                   :
                   N200 M30
                   Jumping back to main program.
N11
N20 G23 N=989
:
N30 M30
```

Explanation

N10: Program N988 is called

N20: Program N989 is called

17. Enable/Disable feed override G25/G26

Purpose

To enable or disable the feed override, in order to control programmed feed movements. If the feed override is disabled, the feed override is fixed to 100%.

Format

To enable feed override:
N... G25

To disable feed override:
N... G26

Parameters

Modal words
H, T1=

Type of function

Modal

Notes and usage

DEFAULT MODE

The CNC system automatically activates G25 at the start of a partprogram.

CANCELLATION

The G26-function is cancelled by G25 or with softkey CLEAR CONTROL or softkey CANCEL PROGRAM or M30.

Example

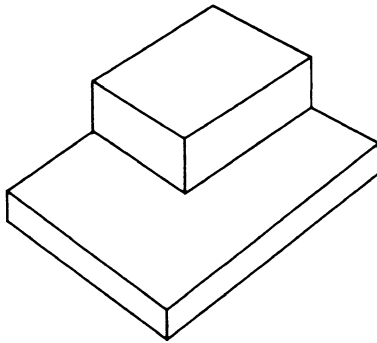
N66 G26 : Feed override switched off, thus fixed to 100%.

N70 G25 : Feed override switched on.

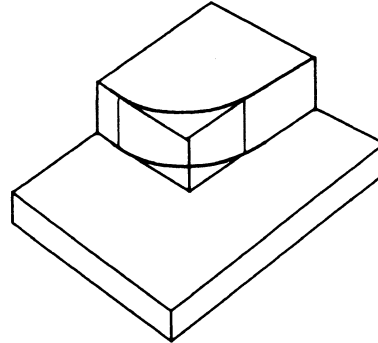
18. Positioning functions G27/G28 active till V320

Purpose

1. For indicating with feed movements (G1,G2/G3,G6) when the next movement starts and if a stop between the movements should occur.
2. For reducing the 'rounding' of corners, caused by positioning lag and the acceleration and deceleration in the axes, when changing feed movement direction. Parameter I3=2 or I3=3 and I7=..
3. For indicating with a rapid movement (G0) if a stop between the movements should occur or must be avoided. Parameter I4=.
4. For switching off and on the positioning logic with G0 movements. Parameter I5=.
5. For feed limitation during circular movements, to reduce the programmed feed rate to keep axes following error to an acceptable minimum and therefore to improve accuracy. Parameters I6=.



Movement without corner rounding



NB9800

Movement with corner rounding

Format

To activate:

N... G28 {I3=...} {I4=...} {I5=...} {I6=...} {I7=..}

To cancel each possibility separately

N... G28 {I3=0} {I4=0} {I5=0} {I6=0}

To cancel all possibilities (default setting):

N... G27

Parameters

I3= 0=off/1=exact
I4= Rapid: 0=exact, 1=rounding
I5= Position logic: 0=with, 1=without

Modal words
H, T1=

Associated functions

G0, G1, G2/G3
F-functions

Type of function

Modal

Notes and usage

Movements

MOVEMENTS WITH INPOSITION

A movement with inposition means, that the next movement starts once all programmed axes have reached their programmed position. A stop occurs between the movements.

MOVEMENTS WITHOUT INPOSITION

A movement without inposition means, that the next movement starts as soon as the interpolator of the CNC has reached the commanded position. No deceleration or axes lag is taken into account. There is no stop between the movements, so the program is executed faster, but with feed movements rounding of corners occur.

Feed movements

FEED MOVEMENTS (G1, G2/G3)

Parameter I3 controls the point at which the next programmed movement starts after a feed movement.

G28 I3=0

The feed movement is executed WITHOUT INPOSITION. There is no stop between the movements and therefore corners are rounded; machining quality is good.

I3=0 is the default setting with feed movements.

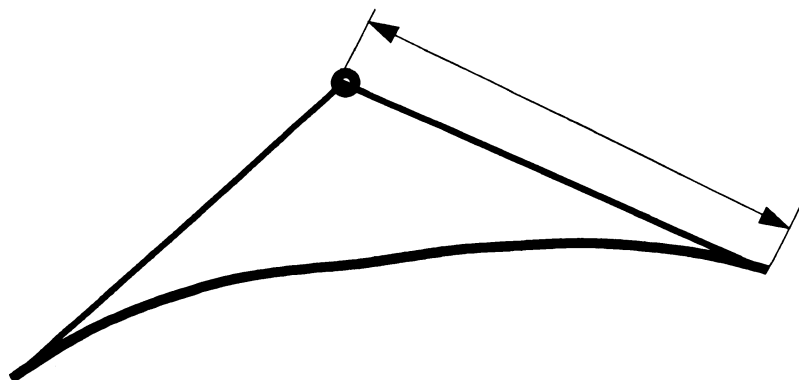
G28 I3=1

The feed movement is executed WITH INPOSITION. A stop occurs between the movements. Corners are sharp, but machining quality is poor.

CORNER ACCURACY WITH LINEAR AND CIRCULAR MOVEMENTS

Rounding of corners are caused by positioning lag and the acceleration and deceleration in the axes, when changing feed movement direction. These rounding effects can be reduced by using the corner accuracy possibility of the control with which either a corner release distance (MC136) or a programmable corner error can be defined.

CORNER RELEASE DISTANCE



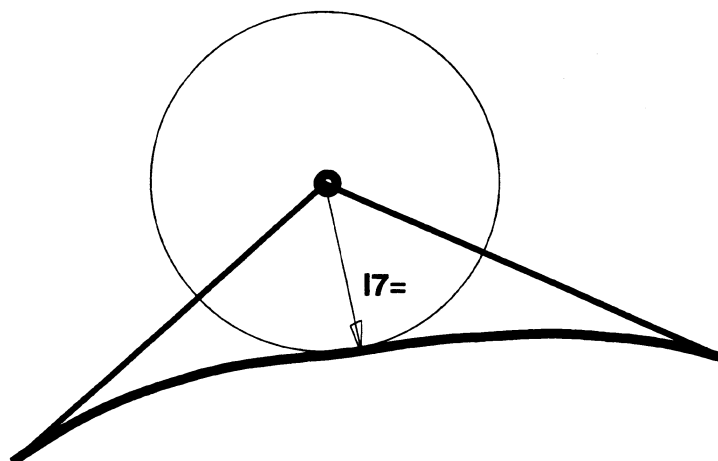
NB9801a

The corner release distance defines a position measured along the axis from the corner. The distance is stored in the Machine Constant Memory. If the axis is between the programmed position and the position defined by the corner release distance, it is assumed that the corner is machined in tolerance. One machine constant for all axes is available.

G28 I3=2

The next movement with a feed movement starts as soon as all axes are between the programmed position and the position defined by the corner release distance. There is no stop between the movements.

Programmable corner error (Only active in a G1 movement)



NB9801b

Defining the corner error

G28 I3=3 I7=..

The allowed corner error, programmed in mm.

G28 I3=3

The feed movement is to be executed taking into account the corner error, programmed with I7=..

The feed rate is automatically reduced by the control to the maximum feed rate with which the corner can be made and the programmed error is not exceeded.

If I7= is not programmed, the value of machine constant (MC137) is used as the maximum corner error.

Remark:

It is only possible to program G28 I3=3 if BTR option (MC262) has been activated.

Note:

1. Corner accuracy can be used with corners between all kinds of linear and circular movements, but not with corners in which splines are involved.
2. A programmed corner error I7=0 will result in a minimal error, which is not necessarily zero due to eg. machine tool characteristics.

Rapid traverse movements

RAPID TRAVERSE MOVEMENTS (G0)

Parameter I4 controls the point at which the next programmed movement starts after a rapid movement (G0).

G28 I4=0

The rapid movement is executed WITH INPOSITION. A stop occurs between the movements.

I4=0 is the default setting with rapid movements.

G28 I4=1

The rapid movement is executed WITHOUT INPOSITION. There is no stop between the movements.

G28 I4=2

The rapid movement is executed, so that the next movement starts once all axes have almost reached their programmed position. This position is defined by the machine constant(MC136) for corner release distance. There is no stop between the movements.

SWITCHING ON/OFF THE POSITIONING LOGIC WITH G0

Parameter I5 indicates if the positioning logic in a G0-block should be executed or switched off. Refer to the function G0 for a description of the positioning logic.

G28 I5=0

G0 is executed with positioning logic.

I5=0 is the default setting.

G28 I5=1

Positioning logic is not active with a G0 movement

Note: The positing logic with a G79 block can not be switched off.

Feed limitation with circular movements**FEED LIMITATION WITH CIRCULAR MOVEMENTS**

Feed limitation is used to reduce the feed of circular movements to a maximum value, so that no contour errors are introduced due to the feed that was programmed. A special formula for feed limitation is used by the control. Parameter I6= is used to indicate which constant should be used in the formula.

G28 I6=0

The feed limitation constant is set by the Machine Tool Control Builder. I6=0 is default setting.

G28 I6=1

Feed limitation constant set to a value stored in the Machine Constant Memory. This machine constant(MC135) is determined by the machine tool builder to best suit the servos and axes drives.

Remark:

Feed limitation (G28, I6=1) only works in G94 (feed in mm/min) and not in G95 mode (feed in mm/rev). So use this function in part programs only if G94 is active.

Cancellation parameter**CANCELLATION ALL PARAMETERS**

All G28 parameters are reset to their default values by programming G27 or by performing the CLEAR CONTROL operation or softkey CANCEL PROGRAMM or M30.

G27 results in G28 I3=0 I4=0 I5=0 I6=0

CANCELLATION OF EACH INDIVIDUAL PARAMETER

Each parameter of the G28 function can be cancelled separately by programming the default setting. The parameters do not influence each other.

Overview

- | | |
|---|----------------|
| 1. G28 without parameter
G1,G2,G3 with In-Position | G28 |
| 2. Movement with feed | |
| G2,G3 without In-Position (initial setting) | G28 I3=0 |
| G1,G2,G3 with In-Position | G28 I3=1 |
| G1,G2,G3 with corner release distance (MC136) | G28 I3=2 |
| G1 with programmable contour accuracy | |
| - Contour accuracy (MC137) | G28 I3=3 |
| - Programmable contour accuracy | |
| I7=.. [0-10000mm] | G28 I3=3 I7=.. |
| 3. Rapid traverse movements G0 | |
| G0 with In-Position (initial setting) | G28 I4=0 |
| G0 without In-Position | G28 I4=1 |
| G0 with corner release distance (MC136) | G28 I4=2 |
| 4. Positioning logic with G0 | |
| G0 with positioning logic (initial setting) | G28 I5=0 |
| G0 with positioning logic | G28 I5=1 |
| 5. Feed limiting for circular movements | |
| G2, G3 with standard value (initial setting) | G28 I6=0 |
| G2, G3 with standard value (MC135) | G28 I6=1 |

19. Positioning functions G27/G28 active from V320

Starting with the V320 software version, the control system features the Look Ahead Feed (LAF) function. Refer to the chapter LOOK AHEAD FEED (LAF) FUNCTION for a description of LAF.

LAF has a lot of influence on the G27/G28 functions.

Purpose

1. For indicating with feed movements (G1,G2/G3,G6) when the next movement starts and if a stop between the movements should occur.
2. For indicating with a rapid movement (G0) if a stop between the movements should occur or must be avoided. Parameter I4=.
3. For switching off and on the positioning logic with G0 movements. Parameter I5=.

Format

To activate:

N... G28 {I3=...} {I4=...} {I5=...}

To cancel each possibility separately

N... G28 {I3=0} {I4=0} {I5=0}

To cancel all possibilities (default setting):

N... G27

Parameters

I3= 0=off/1=exact

I4= Rapid: 0=exact, 1=rounding

I5= Position logic: 0=with, 1=without

Modal words

H, T1=

Associated functions

G0, G1, G2/G3

F-functions

Type of function

Modal

Notes and usage

Movements

MOVEMENTS WITH INPOSITION

A movement with inposition means, that the next movement starts once all programmed axes have reached their programmed position. A stop occurs between the movements.

MOVEMENTS WITHOUT INPOSITION

A movement without inposition means, that the next movement starts as soon as the interpolator of the CNC has reached the commanded position. No deceleration or axes lag is taken into account. There is no stop between the movements, so the program is executed faster, but with feed movements rounding of corners occur.

Feed movements

FEED MOVEMENTS (G1, G2/G3)

Parameter I3 controls the point at which the next programmed movement starts after a feed movement.

G28 I3=0

The feed movement is executed WITHOUT INPOSITION. There is no stop between the movements and therefore corners are rounded; machining quality is good.

I3=0 is the default setting with feed movements.

G28 I3=1

The feed movement is executed WITH INPOSITION. A stop occurs between the movements. Corners are sharp, but machining quality is poor.

Remark: G28 I3= is only possible during G74 movements.

Rapid traverse movements

RAPID TRAVERSE MOVEMENTS (G0)

Parameter I4 controls the point at which the next programmed movement starts after a rapid movement (G0).

G28 I4=0

The rapid movement is executed WITH INPOSITION. A stop occurs between the movements.

I4=0 is the default setting with rapid movements.

G28 I4=1

The rapid movement is executed WITHOUT INPOSITION. There is no stop between the movements.

SWITCHING ON/OFF THE POSITIONING LOGIC WITH G0

Parameter I5 indicates if the positioning logic in a G0-block should be executed or switched off. Refer to the function G0 for a description of the positioning logic.

G28 I5=0

G0 is executed with positioning logic.

I5=0 is the default setting.

G28 I5=1

Positioning logic is not active with a G0 movement

Note: The positing logic with a G79 block can not be switched off.

Cancellation parameters**CANCELLATION ALL PARAMETERS**

All G28 parameters are reset to their default values by programming G27 or by performing the CLEAR CONTROL operation or softkey CANCEL PROGRAMM or M30.

G27 results in G28 I3=0 I4=0 I5=0

CANCELLATION OF EACH INDIVIDUAL PARAMETER

Each parameter of the G28 function can be cancelled separately by programming the default setting. The parameters do not influence each other.

Overview

- | | |
|---|----------|
| 1. G28 without parameter | |
| G1,G2,G3 with In-Position | G28 |
| 2. Movement with feed | |
| G2,G3 without In-Position (initial setting) | G28 I3=0 |
| G1,G2,G3 with In-Position | G28 I3=1 |
| 3. Rapid traverse movements G0 | |
| G0 with In-Position (initial setting) | G28 I4=0 |
| G0 without In-Position | G28 I4=1 |
| 4. Positioning logic with G0 | |
| G0 with positioning logic (initial setting) | G28 I5=0 |
| G0 with positioning logic | G28 I5=1 |

20. Conditional jump G29

Purpose

To jump to a different section of a partprogram (or subprogram) if a parameter is >0.

Other jump conditions like =, <>, >, >=, <, <=, can be programmed when using a relational expression together with the G29-function.

Format

N... G29 E... N=... {K...} {I...}

Parameters

E Jump condition: E > 0
 N= Jump to blocknumber
 K Jump decrement
 I Search direction
 ENNN= Parameter definition

Modal words

E...= Parameter definition.

Associated functions

G14, G22

Type of function

Non-modal.

Notes and usage

BLOCK NUMBER FOR JUMP (N=)

This word specifies the block number of the block to jump to. The block must be in the same partprogram or subprogram.

When there are block numbers with the same number then the first following block with the number is jumped to or a error is shown.

JUMP DIRECTION

A jump can be performed in forward or backward direction in a (sub) program. With I=1 or I=0 the jump is performed forwards. If I=-1 or nothing the jump is performed backwards to the top of the program and the forwards.

JUMP CONDITION (E)

The value of the E-word is used for dictating if a conditional jump must be performed.

If the value of E...>0, a jump is performed.

If the value of E...<=0, the jump is not performed.

JUMP DECREMENT (K)

The K-word allows the value of the E-word to be recalculated and used as the condition for jumping.

If the K-word is not programmed, the E-word's value is reduced by 1 every time a jump is made.

If K0 is programmed, the value of the E-word is not reduced.

If $K > 0$, its value is used to reduce the E-word's value. If eg. K5 were programmed, 5 would be subtracted from the E-word's value after every jump.

If $K < -5$ an error message is displayed.

DEFINING E-PARAMETERS IN A G29-BLOCK

In a G29-block parameters can be defined and calculated. The sequence of executing the block is:

1. load the parameters
2. perform the jump, if the condition is fulfilled

UNCONDITIONAL JUMP

An unconditional jump can be programmed by setting the parameter for the jump condition equal 1.

Eg. G29 E50 N=... E50=1

RELATIONAL EXPRESSIONS

When using relational expressions (see the appendix on E-parameters at the end of this manual), the programming facilities of the conditional jump are substantially expanded.

The relational expression sets the parameter for the jump condition to 0 or 1. The jump is executed as usual.

To keep a program as readable as possible, it is advised to program the relational expression in the same block as the G29. However, the relational expression can be programmed also in a block before the G29, as only the set parameter for the jump condition is used by the G29.

Eg. N.. G29 E1=E2>E3 E1 N=400

This block means:

If the value of E2 is greater than the value of E3, parameter E1 is set =1 and on this setting the jump to N400 is performed.

Example

```
:  
N50 E2=3  
N51  
:  
N100 G29 E2 N=51
```

Explanation:

N50: Set initial value of the parameter E2 to 3.

N100: When $E2 > 0$, jump to block number 51 and then continue to execute the program blocks in sequential order till block N100.

At each jump parameter E2 is decremented automatically by 1. Therefore, after 3 loops, parameter E2 is equal to zero and no more jumps executed. The program continues after block N100 in the sequential order.

21. Activate/ deactivate compensation (active in V320) G39

Purpose

Programmed contours can be changed by an offset.

Format

Activate offset:

N... G39 {R...} {L...}

R: Tool radius offset

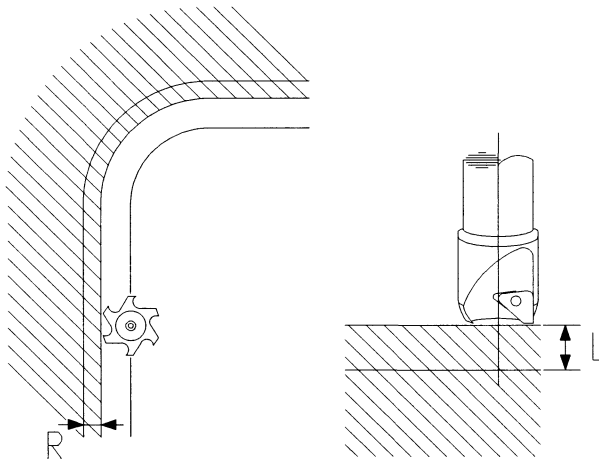
L: Tool length offset

Deactivate tool length offset:

N... G39 L0

Deactivate tool radius offset:

N... G39 R0



Parameters

L	Toollength offset
R	Toolradius offset

Associated functions

G17, G18, G19, G40, G41, G42, G43, G44, G61, G62, G66, G67, G87-G89, G141, G182, G201, M6, M66, M67

Type of function

Modal

Notes and usage

Tool length offset:

The tool length offset operates into the direction of the tool axis. Tool length offset changes will become effective with the next feed movement.

Tool radius offset:

The tool radius offset operates in the machining plane, but is only effective with an active cutter radius compensation.

If the cutter radius compensation is inactive, tool radius offset changes will become effective after the cutter radius compensation (G41/G42, G43/G44) has been activated.

If the cutter radius compensation is active, tool radius offset changes will be corrected in the next movement block linearly over the entire path.

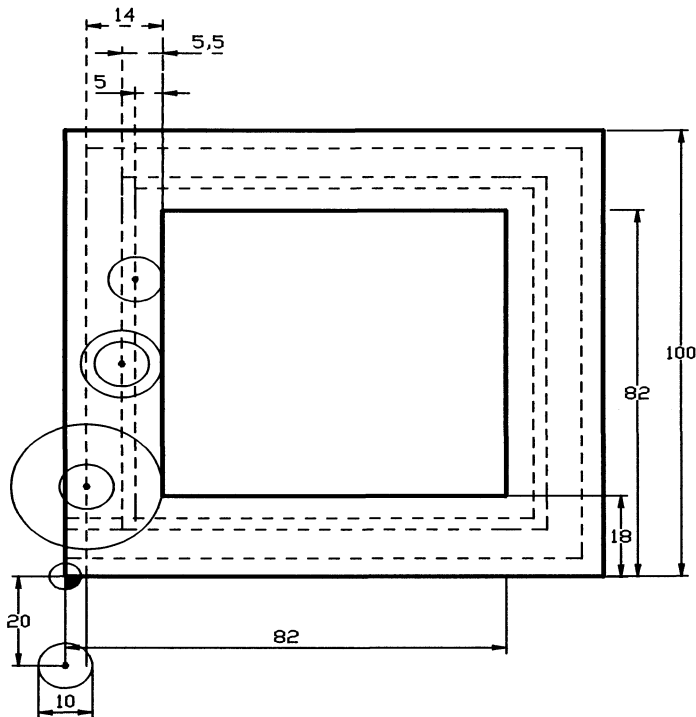
Offset programming is maintained after a tool change (M6, M66) or change of plane (G17, G18, G19).

Note:

There will be a radius offset override when the following functions are activated: G6, G83-G89, G141, G182. The length offset remains effective. Offset programming should be deactivated before these functions.

Example

Rectangular milling by roughing (2x) and finishing (1x)



```

N39001
N1 G98 X-10 Y-10 Z10 I120 J120 K-60
N2 G99 X0 Y0 Z0 I100 J100 K-40
N3 T1 M6
N4 G39 L0 R9
N5 F500 S1000 M3
N6 G0 X0 Y-20 Z5
N7 G1 Z-10
N8 G43 X18
N9 G41 Y82
N10 X82
N11 Y18
N12 X0
    
```


N13 G40
N14 G39 R0.5
 N15 G14 N1=8 N2=13
N16 G39 R0
 N17 G14 N1=8 N2=13
 N18 G0 Z10
 N19 M30

Explanation:

N1 Define graphic window
 N2 Define material
 N3 Change tool (cutter diameter 10 mm)
 N4 Activate tool radius offset. (Cutter radius for radius compensation is $(5+9 =) 14$ mm)
 N5 Activate feed and spindle speed
 N6 Approach starting position
 N7 Moving to depth
 N8 Approach contour with radius compensation
 N9 Initial roughing of the rectangle. Offset is 9 mm.
 N13 Turn off radius compensation.
 N14 Change tool radius offset. (Cutter radius for radius compensation is $(5+0.5 =) 5.5$ mm)
 N15 Repeat rectangle (second roughing operation). Offset for finishing is 0.5 mm
 N16 Change tool radius offset. (Cutter radius for radius compensation is 5 mm)
 N17 Finish the rectangle.
 N18 Approach clearance distance
 N19 Program end.

Activate/ deactivate compensation (active in V320) G39

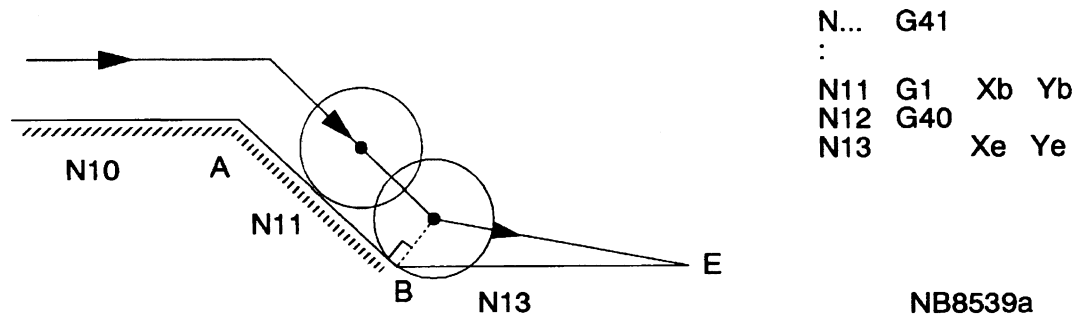
22. Cancel tool radius compensation G40

Purpose

To cancel radius compensation. The tool now moves along the programmed path on the workpiece.

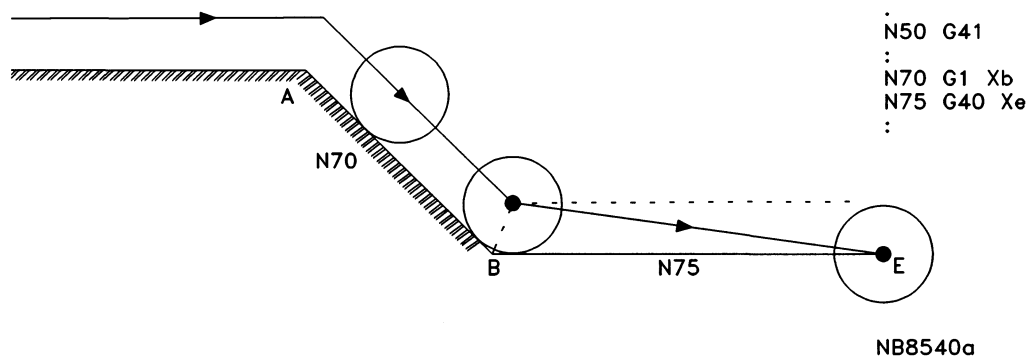
Format

N... G40 {axis coordinates}



N... G40 {without movement}

The radius compensation LEFT is active from point A to point B. At point B, radius compensation is cancelled and programmed movements refer to the tool point.



N... G40 {with movement}

Radius compensation LEFT is active from point A to point B. At point B, radius compensation is cancelled and programmed movements refer to the tool point.

Associated Functions

G41/G42, G43/G44, G61/G62, G141

Type of function

Modal: (G41 to G44)

Notes and usage

DEFAULT MODE

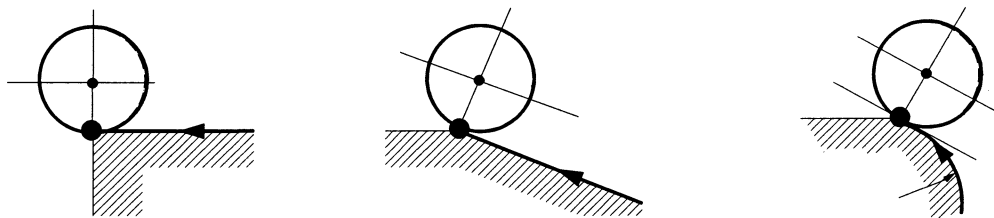
G40 is made active automatically when the CNC system is switched on, M30 and softkeys CLEAR CONTROL and CANCEL PROGRAM operations are performed.

AXES COORDINATES

It is advised to program G40 in a separate block without axis coordinates.

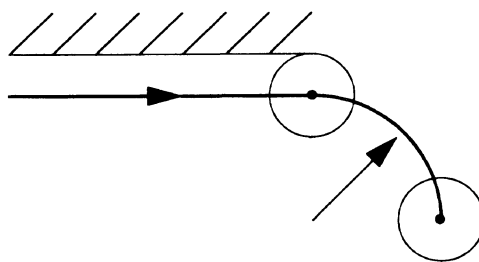
(a) G40 without axis coordinates.

In this case, the previous line or circle will be cut completely. The centre of the tool will be moved to a position perpendicular to the contour in the previous endpoint. The compensation is switch off in the next movement.



NB9645

G40 followed by a linear movement



```

N... G41
:
:
N... X...
N... G40
N... G2 X... Y... R...
    
```

NB9659

G40 followed by a circular movement

If the G40-block is followed by a circular movement, an arc with the programmed radius is inserted between the corrected endpoint of the G40-block and the programmed tool tip position in the block with the circle

(b) G40 with axis coordinates.

The start point of the movement is calculated with full compensation. During the movement the compensation is switched off and the endpoint is without compensation. This way of programming can be used when the compensation can be switched off without damaging a contour.

(c) Tangential exit.

It is also possible to leave a contour with G62.

N.. G62 X.. Z.. R.. (F)

LIFTING THE TOOL FROM THE PLANE OF OPERATION

The tool can be lifted from the plane of operation with

(a) a linear movement

If the plane of operation is e.g. the XY-plane and the tool is lifted by programming a linear movement in YZ, the tool correction in the third axis (X) is cancelled too. So a movement takes place in three axes simultaneously.

(b) a circular movement

If the plane of operation is e.g. the XY-plane and the tool is lifted by programming a circular movement in YZ, the tool correction in the third axis (X) is not cancelled. So only the circular movement in the YZ-plane is executed.

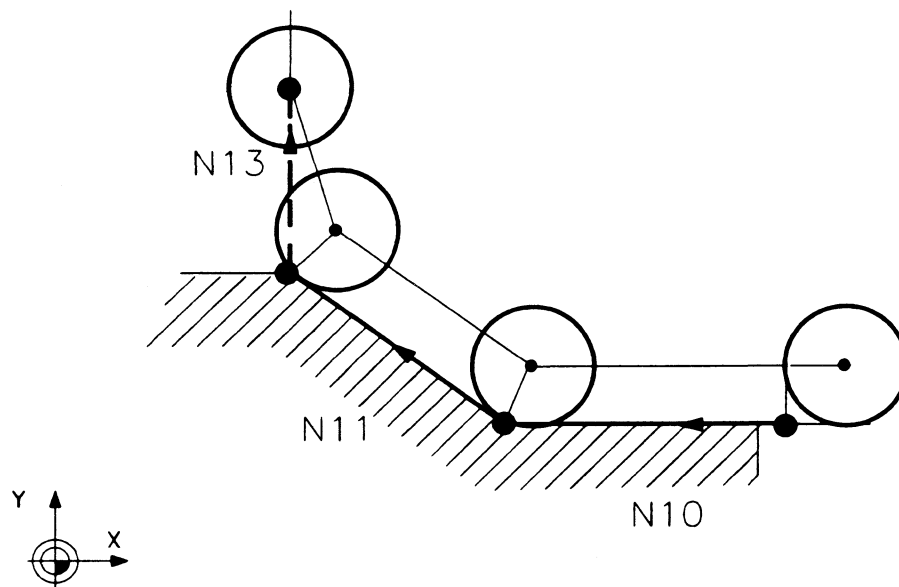
Note:

The start position of the circle is in the Y-axis the corrected position and in the tool axis (Z) the depth of operation.

Programming subsequential identical axes positions in G43 and G40 may cause a positioning error in the axis concerned over the toolradius. This can be prevented by programming in G40 an axis position slightly different (e.g. 1 micron) from the previous position in G43.

Example: N100 G43 X10

Example



NB9660

```

N9 G42
N10 G1 X..
N11 X... Y...
N12 G40
N13 G0 Y...
    
```

Explanation:

<p>N9:</p> <p>N10-11:</p> <p>N12:</p> <p>N13:</p>	<p>activate radius compensation on right side of contour</p> <p>move tool to programmed coordinates. Include tool radius into calculations.</p> <p>cancel radius compensation.</p> <p>move tool from the previous compensated position to the uncompensated endpoint of this rapid movement.</p>
---	--

23. Tool radius compensation (left/right) G41/G42

Purpose

To allow for workpiece dimensions to be programmed rather than the tool path. The tool path is automatically calculated by the CNC to be a path parallel to the programmed workpiece contour.

G41 activates radius compensation LEFT of the workpiece

G42 activates radius compensation RIGHT of the workpiece

In both cases when looking in the same direction as the movements of the cutting tool.

Formats

N.. G41/G42 {axis coordinates}

Associated functions

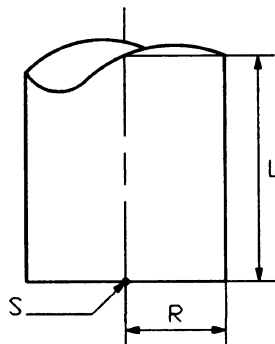
G40, G43/G44, G61/G62 and for 3D-tool correction G141

Type of function

Modal

Notes and usage

POSITION OF TOOL POINT



NB9646

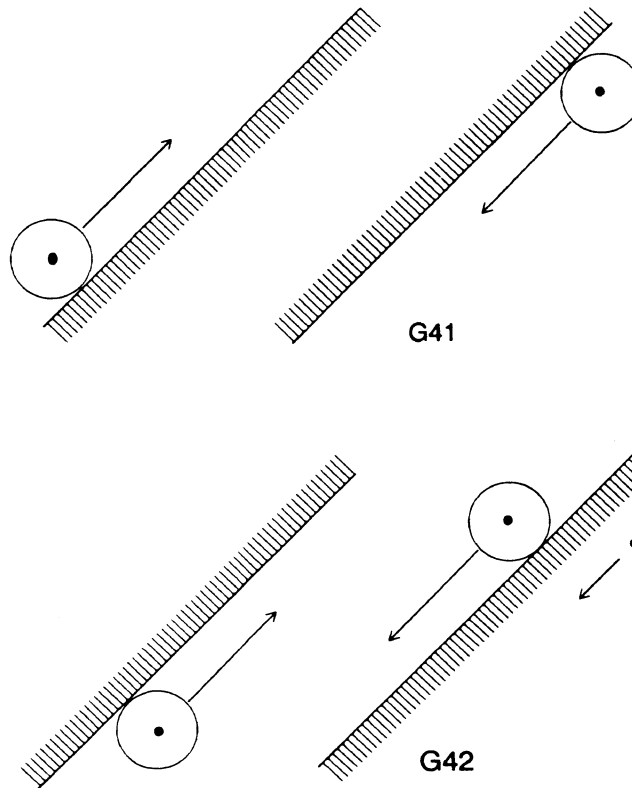
S = tool point specified by the tool dimensions L and R

TOOL MEMORY

The radius compensation function uses the tool radius from the tool memory.

COMPENSATION LEFT AND RIGHT

When using radius compensation the CNC must know whether the tool is cutting at the left or right side of the work piece. The function G41 or G42 is used for this purpose.



NB9648

To decide which function must be programmed, it is necessary to look in the same direction as the movement of the cutting tool. If the tool moves on the left of the workpiece surface, G41 is used and on the right, G42.

This method assumes that a positive radius value is stored in the tool memory, when the program is executed.

However, if the stored radius value is negative, the following applies:

G41 and negative radius = G42 and positive radius

G42 and negative radius = G41 and positive radius

Refer to TOOL RADIUS CORRECTION for using negative radius values in the tool memory.

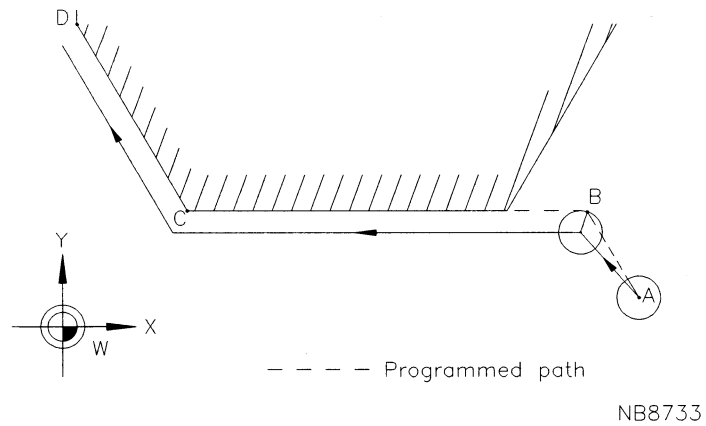
STARTING RADIUS COMPENSATION

There are three options to start radius compensations:

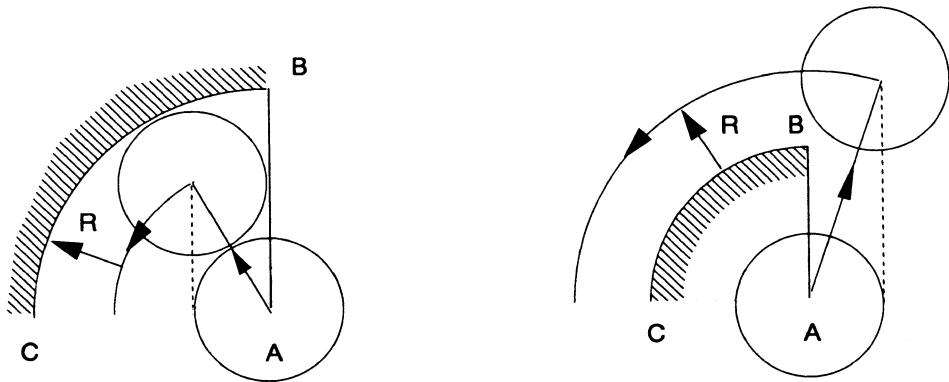
- 1) direct with the functions G41/G42.
- 2) With the functions G43/G44.(See G43)
- 3) Or the function tangential approach.(See G61)

The part programmer must ensure that the tool does not collide with the workpiece, when radius compensation is being started. The start point must therefore be at a safe distance outside the workpiece.

When G41 or G42 is used, the intersection point between two related contour elements is calculated by the CNC and the tool moves to this point.

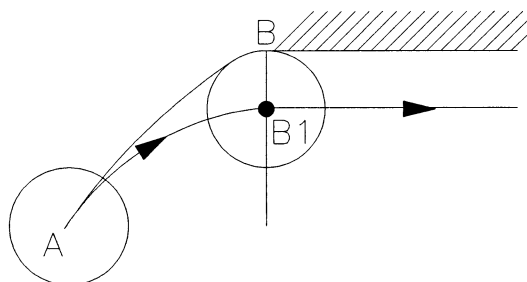


Activate radius compensation G41/G42 with line to line.



Activate radius compensation G41/G42 with line to circle.

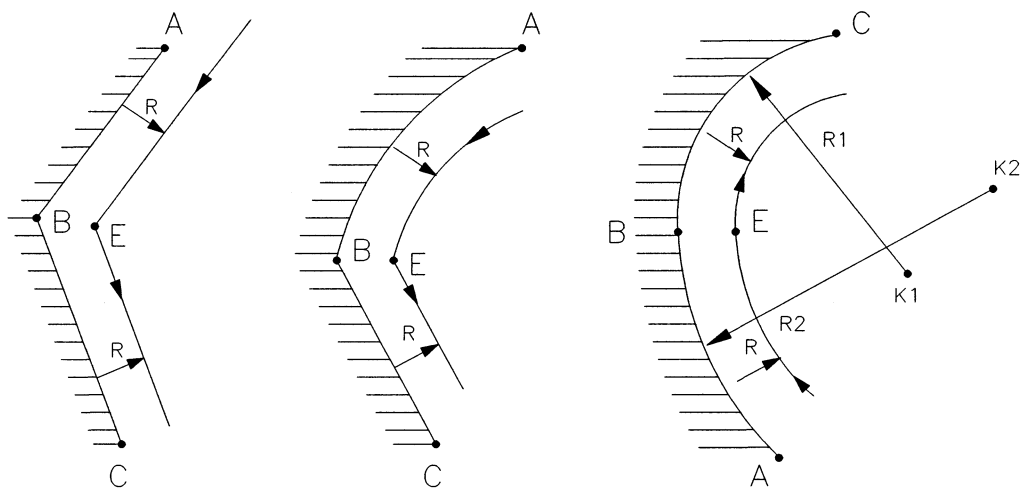
If radius compensation is activated during a circular movement (AB), the tool moves with a circular arc from the point the tool is standing (A) to the first calculated position (B1).



Activate radius compensation G41/G42 on a circle

INTERNAL CONTOURS

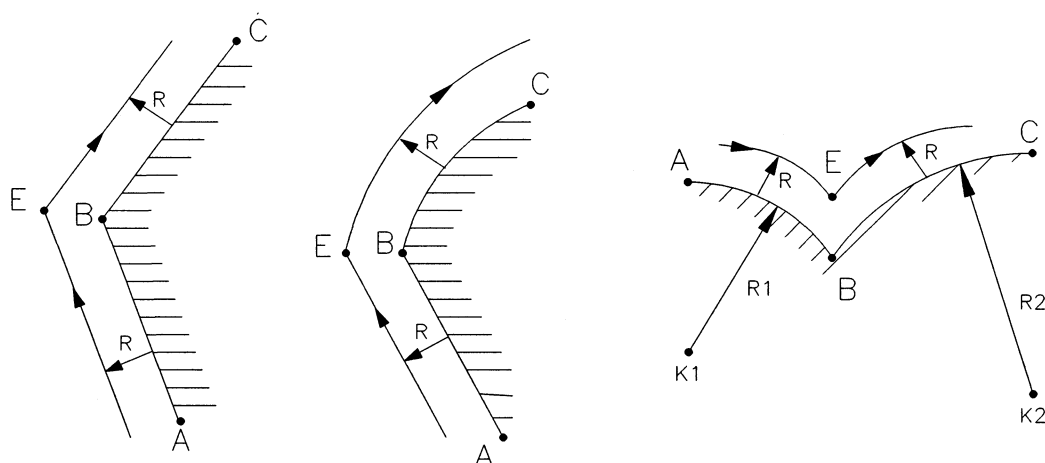
When radius compensation is used, the tool path is always the same distance from the programmed contour, except at intersection points between contour elements. These points are calculated by the control automatically.



NB8715

EXTERNAL CONTOURS

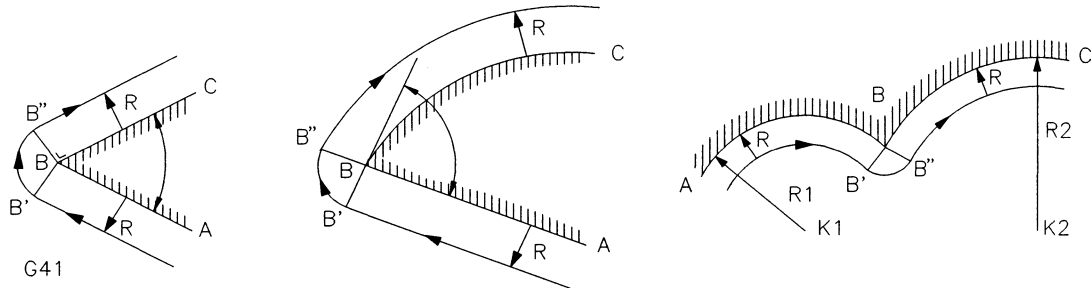
Intersection points between external contour elements are calculated and the tool moves to that position, whenever the angle between the elements is greater than a Machine Constant Value.



NB8716

EXTERNAL CONTOURS WITH SHARP CORNERS

If the angle between two external contour elements is less than the Machine Constant Value(MC711), a circular movement between the two elements is generated by the CNC. This circular movement is treated as part of the previous block. Therefore, if a SINGLE BLOCK operation is commanded, the tool stops after this circular movement.



NB9115

TOOL RADIUS CORRECTION

In general a NC programming system calculates the tool path taking into account the radius of a nominal tool. The radius compensation as described, allows to use a real tool for machining the part and to use a deviation on the radius of the nominal tool to let the control calculate the path of the actual tool.

A correction value on the tool radius including a sign, is therefore stored in the tool memory.

"+ correction value": for an oversized cutter, thus with a radius greater than the radius of the nominal tool.

"- correction value": for an undersized cutter

MOVEMENTS

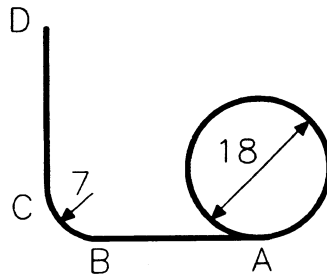
Programming either G41 or G42 does not result in a preparatory function for a motion (G0, G1, G2 or G3) also being activated. The last programmed function for a movement remains active.

PROGRAMMING ERRORS

If the tool radius is too large, the workpiece might be damaged in some situations.

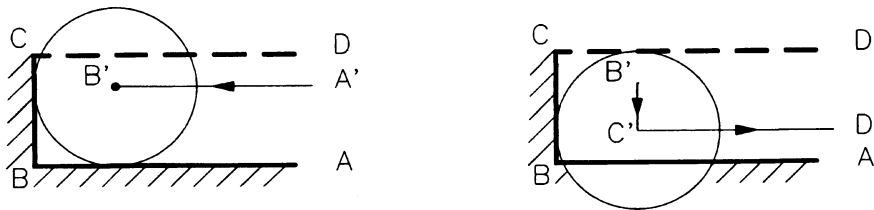
- a. The radius of the tool is equal to or larger than the workpiece radius. If this occurs an error message is generated.

Note: the tool radius must be at least 0.001 mm (0.0001") smaller than the programmed radius.



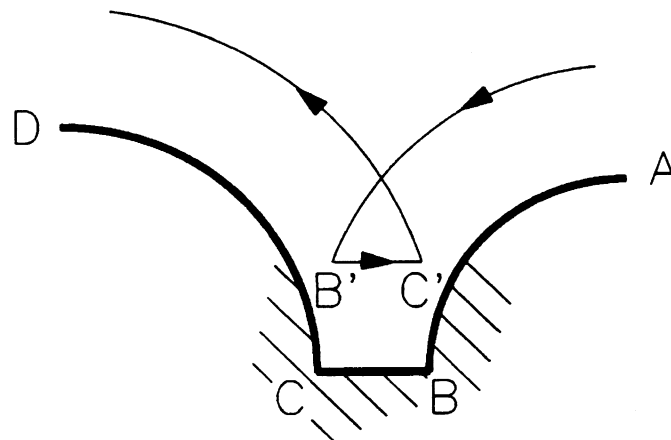
NB6250a

- b. The contour from AB to BC is programmed. With active radius compensation the tool is retracted along CD. If BC is smaller than two times the tool radius, the tool collides with the workpiece during the movement from B' to C' and from C' to D'.



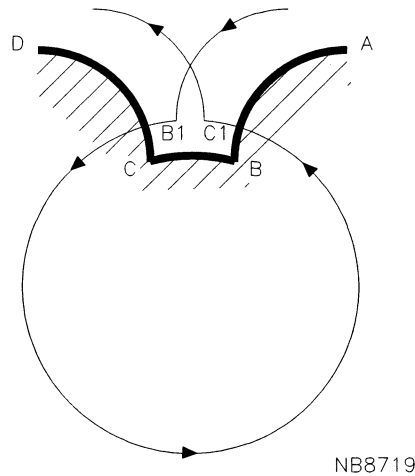
NB9620

- c. A contour of the shape given in the illustration below, is programmed. If the straight line is smaller than two times the tool radius, the tool collides with the workpiece during machining.



NB6250c

- d. A contour of the shape given in the illustration below, is programmed. The tool moves to point B1, then from B1 to C1 and then parallel along CD. The movement from B1 to C1 takes place in the same direction as programmed on the circle BC. If the circular movement BC is too small, this results in the tool making almost a complete circle before it arrives at C1.

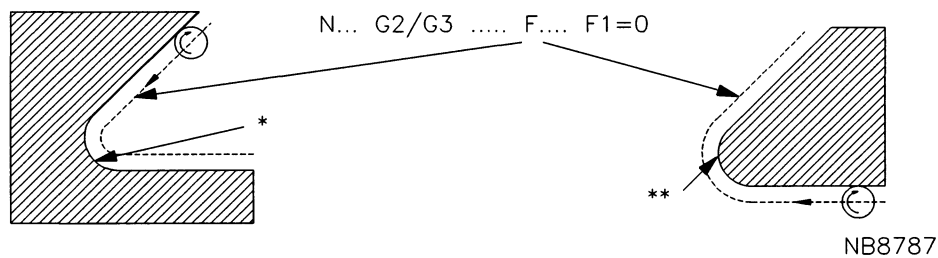


CONSTANT CUTTING FEED

The parameter 'F1=.' is used to ensure that the programmed feed rate along a workpiece contour remains constant regardless of the radius of the mill and the contour shape. This controlled velocity is called the **CONSTANT CUTTING FEED**.

- F1=0 C.C.F. not applied (default mode; also set at CLEAR CONTROL or M30 or softkey CANCEL PROGRAM).

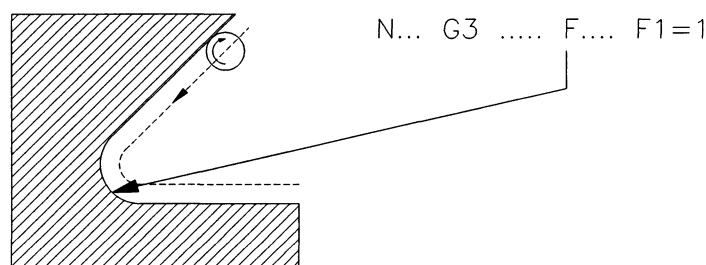
The programmed feed rate should be the velocity of the tool tip.



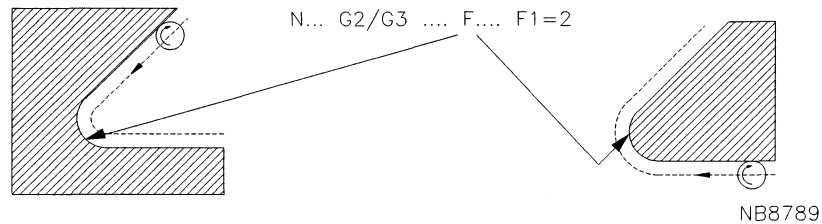
* Cutting feed too high

** Cutting feed too low

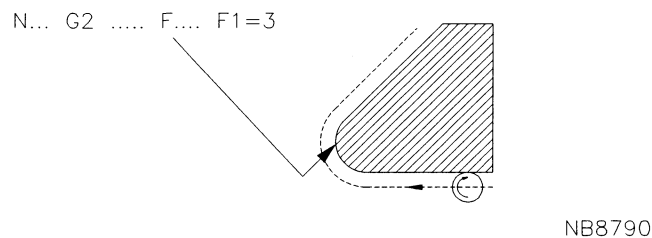
- F1=1 C.C.F. applied only on the inside of arcs. The programmed feed rate is reduced to assure that the tool tip moves with the reduced velocity on the inside of an arc.



- F1=2** C.C.F. applied on the inside and outside of arcs. The programmed feed rate is reduced (inside arc) or increased (outside arc) to assure that the tool tip moves with the recalculated velocity. If the increased velocity is greater than the maximum feed rate (a Machine Constant value) the maximum feed rate is used.



- F1=3** C.C.F. applied only on the outside of arcs. The programmed feed rate is increased to assure that the tool tip moves with the increased velocity on the outside of an arc. If the increased velocity is greater than the maximum feed rate (a Machine Constant value) the maximum feed rate is used.



SWITCHING FROM ONE RADIUS COMP. FUNCTION TO ANOTHER ONE

When switching from one function, e.g. G41 to G42, G43 or G44, the tool ends in a position which is calculated with the first function active and starts in a position calculated with the other function active. When these two positions do not coincide a linear feed movement from one position to the other one is executed.

ENDING RADIUS COMPENSATION

The function G40 cancels the tool radius compensation. Thereafter programmed coordinates refer to movements of the tool tip.

PLANE FOR RADIUS COMPENSATION

The radius compensation is performed in the plane indicated by G17, G18 and G19.

TOOL AXIS MOVEMENT

A simultaneous movement of the tool axis and the axes of the main plane (defined with G17, G18 or G19) with activated radius compensation is possible. The compensated movements in the main axes can be linear or circular.

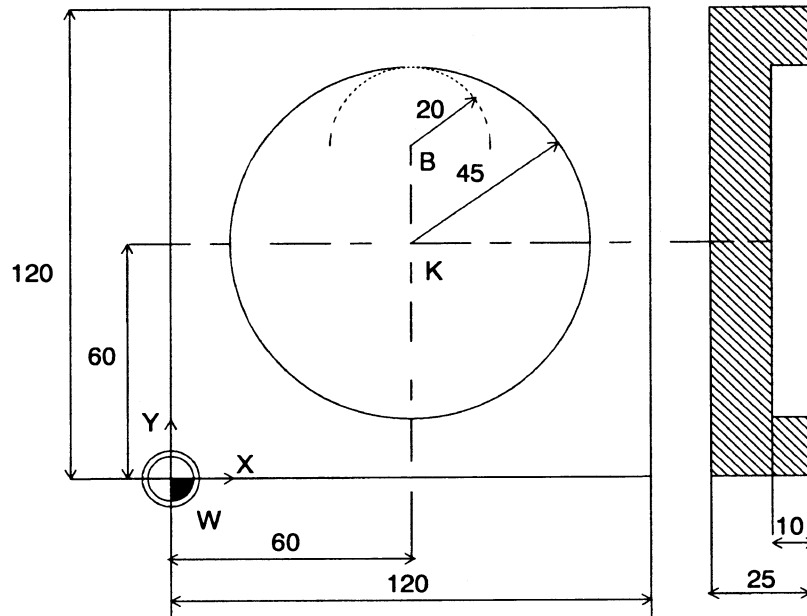
HELIX INTERPOLATION

If helix interpolation is used, radius compensation can be used on the circular movement in the plane indicated with G17, G18 or G19.

USING THE CYLINDRICAL COORDINATE SYSTEM

When the cylindrical coordinate system (G182) is activated, the functions G41/G42/G43/G44 as described can also be used in the plane of the cylinder.

EXAMPLE 2.



NB8603

```

N9998
N1 G17
N2 G54
N3 T1 S3000 M6
N4 G0 X60 Y85 Z0 M3
N5 G1 Z-10 F500
N6 G43 X80 F300
N7 G41
N8 G3 X60 Y105 R20
N9 I60 J60
N10 X40 Y85 R20
N11 G40
N12 G0 Z200 M30

```

Explanation:

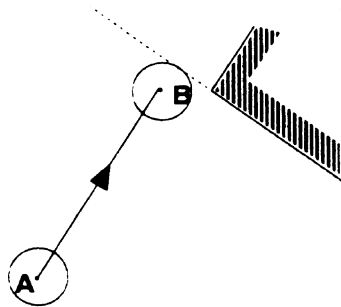
N3: Load the tool. The mill has a diameter of 10 mm.
 N4: Start the spindle and move tool to starting point B.
 N5: Feed the tool to depth.
 N6: Move the tool to the starting point of the small circle. Set new feed rate to 300 mm/min.
 N7: Set radius compensation LEFT.
 N8: Move the tool with a circular movement, to enter the contour.
 N9: Mill the complete circle.
 N10: Exit the contour by using a small circular movement.
 N11: Cancel the radius compensation.
 N12: Retract the tool out of the workpiece. End of program

24. Tool radius comp. to/past endpoint G43/G44

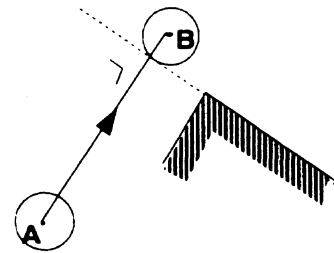
Purpose

To move the tool with cutter radius compensation. TO/PAST a programmed position.

- G43 activates radius compensation 'TO' a programmed position (tool radius is subtracted from the programmed position).
- G44 activates radius compensation 'PAST' a programmed position (tool radius is added to the programmed position).



G43 'TO'



NB9647

G44 'PAST'

Format

N... G43/G44 {axis coordinates}

Associated functions

G40, G41/G42, G61/G62 and for 3D-tool correction G141

Type of function

Modal

Alternative

G43 is mostly used for axis parallel positioning movements. If the positioning movement is not parallel to the axis, starting point B should be calculated. It is therefore wiser to use the possibilities of G61 (tangential approach).

Notes and usage

CIRCULAR MOVEMENT

If a G43 or G44 is active with a circular movement (G2/G3), an error message is displayed.
With circular movements G41 or G42 must be used.

AXIS PARALLEL MOVEMENT

If a G43 or G44 is used with an axis parallel movement and only one coordinate is programmed, the position in that axis is calculated. The other axis remains unchanged. So G43/G44 only can be used with radius compensation and axis parallel movements.

ENTERING A CONTOUR WITH G43 OR G44

The functions G43 and G44 can be used to enter a contour on the normal (= perpendicular to) of any contour element. This way of entering a contour is recommended because there is less change of accidental collision between tool and workpiece than the direct use of the functions G41 or G42. See also STARTING RADIUS COMPENSATION with G41/G42.

SWITCHING FROM ONE RADIUS COMP. FUNCTION TO ANOTHER ONE

When switching from one function, e.g. G41 to G42, G43 or G44, the tool ends in a position which is calculated with the first function active and starts in a position calculated with the other function active. When these two positions do not coincide a linear feed movement from one position to the other one is executed.

ENDING RADIUS COMPENSATION

The function G40 cancels the tool radius compensation. Thereafter programmed coordinates refer to movements of the tool tip.

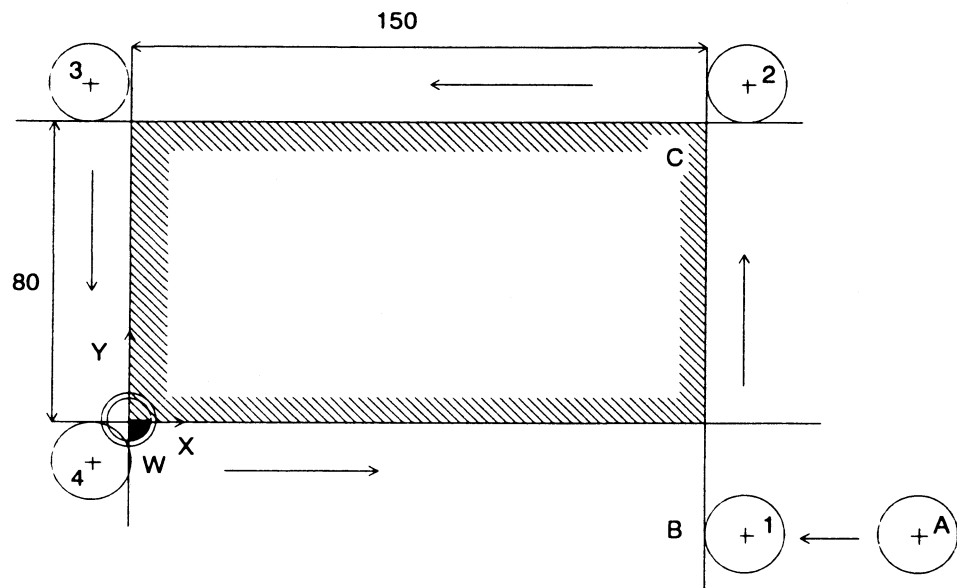
PLANE FOR RADIUS COMPENSATION

The radius compensation is performed in the plane indicated by G17, G18 and G19.

USING THE CYLINDRICAL COORDINATE SYSTEM

When the cylindrical coordinate system (G182) is activated, the functions G41/G42/G43/G44 as described can also be used in the plane of the cylinder.

EXAMPLE 2.



NB8602

N9 T1 M6
N10 G0 X200 Y-20 Z-5 S1000 M3
N11 G43 X150
N12 G1 F200
N13 G44 Y80
N14 X0
N15 Y0
N16 X150
N17 G40 Y-20
N18 G0 X200

Explanation:

- N9: Load tool 1 and its offsets.
N10: Make spindle rotate clockwise at 1000 rev/min, move the tool to position A and then at depth.
N11: Move the tool rapidly to point B.
N12: Set linear feed rate to 200 mm/min.
N13: Move the tool along the Y-axis PAST edge Y80 (point 2). The function G44 remains active in the blocks that follow.
N14: Move the tool along the X-axis PAST edge X0 (point 3).
N15: Move the tool along the Y-axis PAST edge Y0 (point 4).
N16: Move the tool along the X-axis PAST edge X150. The tool is free from the part
N17: Cancel the radius compensation.
N18: Rapid traverse movement to position A.

25. Axis parallel measuring movement G45

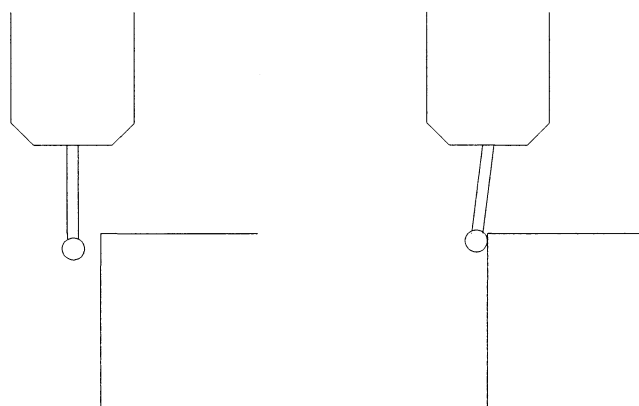
Note

Use of this function is limited only to programs made on earlier control systems.

The G45 function operates only parallel to the axis. G145 has improved functionality and is also able to perform measurements which are not parallel to the axis. It is therefore wiser to use the new G145 basic measurement movement.

Purpose

To measure the actual axis coordinate with a touch trigger probe when being moved in the axis to the programmed position. This allows the difference between the actual and programmed position to be used to check the dimensional accuracy of the workpiece.



NB8258

Probe moves towards the workpiece

Probe is 'triggered'

Format

N... G45 [Measuring position] {I+/-1} {J+/-1} {K+/-1} {L+/-1} {X1=...} {N=...} {E...}

The plane for the rotary table is determined by the definition of the 4th axis in the machine constant list. (MC117 should be 4 and MC118 should be B(66) or C(67)). L relates to the 4th axis B or C. Rotary axis A is not allowed.

Parameters

Measuring position

X,Y,Z Measurement target coordinate

C Measurement target angle

P Point definition number

Measuring parameters

I Measurement direction for X axis

J Measurement direction for Y axis

K Measurement direction for Z axis

L Measurement direction rotary-axis

X1= Measurement path length

Measuring results

E Parameter-nr measured coordinate

N= Point-nr.for measured coordinate

For absolute and incremental programming

X90=,Y90=,Z90= Absolute measurement target

A90=,B90=,C90= Abs. measurement target angle

X91=,Y91=,Z91= Incremental measurement target

A91=,B91=,C91= Incr. measurement target angle

The difference between the measured and programmed coordinate is calculated and stored internally for use with G49 or G50.

Modal words

E...=

Associated functions

G46, G49, G50 and the basic measuring movement G145

M25, M27, M28

Wordwise absolute/incremental programming (X90=..., X91=..)

Type of function

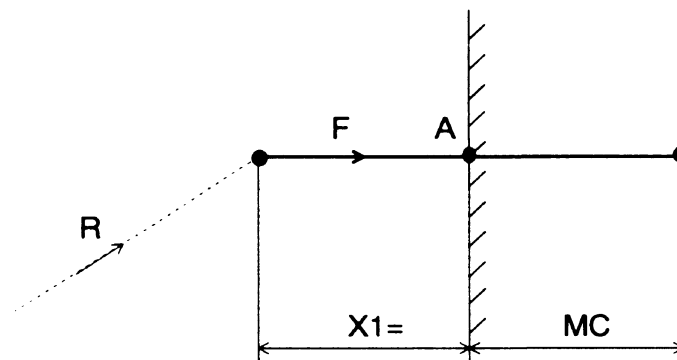
Non-modal.

Notes and usage

MEASURING POSITION

The programmed coordinates specify the point to be measured.

PRE- AND POST-MEASURING DISTANCE (X1=)



NB6901

The pre-measuring distance defines the position in the axis to be measured from where the movement with the measuring feed starts. This distance is programmed with the word X1=.

If X1= is not programmed, a Machine Constant value(MC844) is used.

With the post-measuring distance (MC845) is defined how far the probe can pass the programmed position of the axis before it is triggered.

MEASURING SEQUENCE

1. The probe moves rapidly to the pre-measuring position which is defined by the programmed position and the pre-measuring distance in the axis to be measured. This movement is executed with the positioning logic of G0.
2. After the probe reaches the pre-measuring point it moves at a fixed feed rate (MC843) along the indicated axis in the programmed direction towards the programmed position in the axis. The probe can pass this point, but must be triggered along the path between the pre- and post-measuring distance.
3. When the probe touches the workpiece, the measured coordinate is stored and the probe rapidly moves back to the pre-measuring position.

STORING MEASURING RESULT (E, N=)

The measured coordinate can be stored in either the E-Parameter Memory (E) or and the Point Memory (N=).

Storing the coordinate in an E-parameter has the advantage of allowing additional calculations to be performed, such as in a macro.

The difference between the measured and programmed coordinate is calculated and stored internally for use with G49 or G50.

The stored differences are cancelled as soon as a new measuring function (G45 or G46) is activated or with softkey CLEAR CONTROL or CANCEL PROGRAM.

ERROR MESSAGES

An error message is displayed and the movement stops,

1. if the probe touches an obstruction during the rapid movement to the pre-measuring position,
2. if the probe exceeds the post-measuring distance.

COLLISION PROTECTION

As soon as the measuring probe is triggered during any other movement than the actual feed movement for the measurement itself, an error is generated and the movement interrupted. Sometimes the probe is triggered due to very fast movements and not by a real collision.

- A machine constant (MC850) can be used to store the information that collision protection
- is switched off during the measurement movement and possibly during retraction after measurement;
 - is effective during all movements or only during feed movements.

TOOL MEMORY

The radius of the probe and its length are stored in the tool memory together with a tool number. Tool type Q3=9999 can be entered to indicate the measuring probe.

Example: P5 T5 Q3=9999 L150 R4

When tool T5 is called with Q3=9999 the control system recognizes this tool as the measuring probe. The probe radius is called and used to correct the measurement position.

If a function for spindle direction (M3 or M4) is entered, this function is suppressed and an error message displayed.

AIR BLOW BEFORE MEASURING

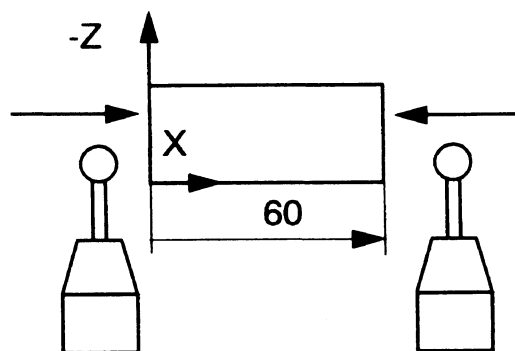
To clean the workpiece at the position to be measured an air blow can be executed during a fixed time (MC842).

The air blow is activated once by a M-function and executed each time a pre-measuring position is reached. Refer to the machine tool builder's documentation for the number of the M-function for the air blow.

RESTRICTION

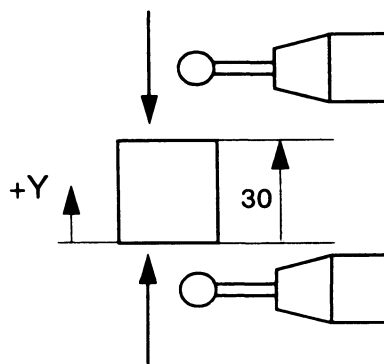
1. Only one axis coordinate can be measured in a G45-block.
2. An omnidirectional probe should be used.
3. In the tool axis the probe is only triggered if pressed. This means that a measurement in the positive direction of the tool axis is not possible.

Note: The G45 function is also used in conjunction with the M25 function for measuring tool dimensions.
Refer to G45 + M25 section for additional information.

Example**EXAMPLE 1: Measuring a point in X-axis**

Measuring in positive direction
 N.. G45 X0 Y20 Z-10 I1 E1 N=1

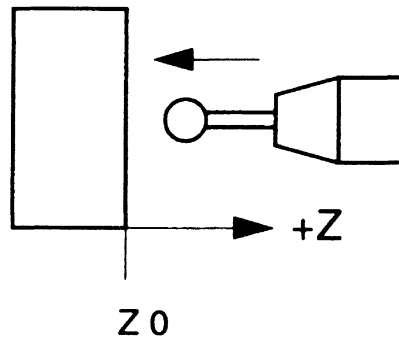
Measuring in negative direction
 N.. G45 X60 Y20 Z-10 I-1 E1 N=1

EXAMPLE 2: Measuring a point in Y-axis

Measuring in positive direction
 N.. G45 X30 Y0 Z-10 J1 E1 N=1

Measuring in negative direction
 N.. G45 X30 Y30 Z-10 J-1 E1 N=1

EXAMPLE 3: Measuring a point in Z-axis



Measuring in negative direction
N.. G45 X30 Y30 Z0 K-1 E1 N=1

Note: Measuring in the tool axis is only possible in the negative direction.

Explanation of the examples:

The point is measured, the measured position calculated and stored in Point Memory location 1 and in parameter E1.

26. Measure tool dimensions G45+ M25

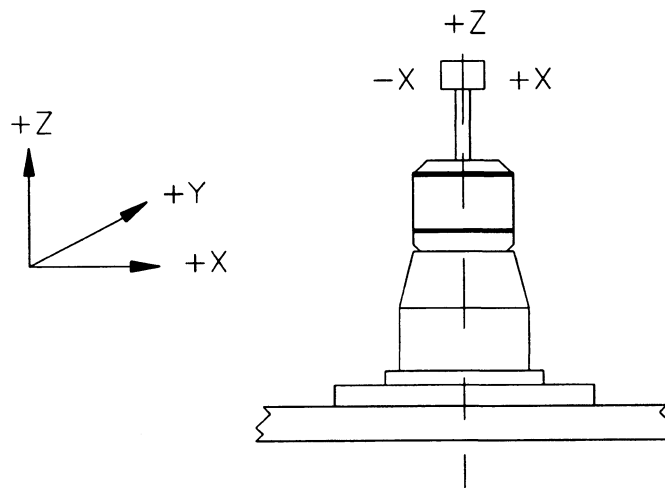
Note

Use of this function is limited only to programs made on earlier control systems.

The G45 function operates only parallel to the axis. G145 has improved functionality and is also able to perform measurements which are not parallel to the axis. It is therefore wiser to use the new G145 basic measurement movement.

Purpose

To measure tool dimensions using a square-head measuring probe.



NB9806

Format

N... G45 {I...} {J...} {K...} {X1=...} M25

Parameters

I Measurement direction for X axis
 J Measurement direction for Y axis
 K Measurement direction for Z axis
 L Measurement direction rotary-axis
 X1= Measurement path length

Modal words

E...=

Associated functions

G45, G46, G49, G50
 M26, M27, M28

Type of function

Non-modal.

Notes and Usage

MEASURING TOOL DIMENSIONS

A square-head measuring probe mounted at a fixed position on the machine tool, is used for measuring the tool dimensions.

Measuring in the tool axis gives the tool length.

Measuring in two directions of the same axis gives the tool radius.

POSITION OF THE SQUARE-HEAD PROBE

The position of the square-head measuring probe (MC3155, MC3755) and its width are red in the Machine Constant Memory(MC847).

MEASURING SEQUENCE

The measurements are executed in the same way as with G45. In stead of programming the position of the fixed probe, its coordinates are picked up from the Machine Constant Memory.

UPDATING THE TOOL MEMORY

The tool memory is updated with the function G50. Refer to that function for details.

Note

1. Tool measurement can also be performed on the control in the mode OPERATE. Refer to the Operating Manual for details.
2. Refer to G145 for an example for automatic measurement of the tool dimensions with a measuring box.

Example

Measuring tool length

```
N89 T1 M6  
N90 G45 K-1 X1=5 M25  
N91 G50 T1 L1=1
```

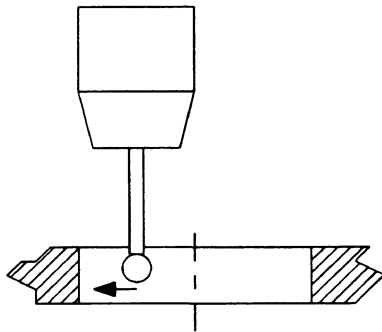
Explanation:

N89: Load tool 1
N90: Measure the tool length in the negative direction of the Z-axis. Pre-measurement distance is 5mm.
N91: Correct the tool length of tool 1 in the tool memory

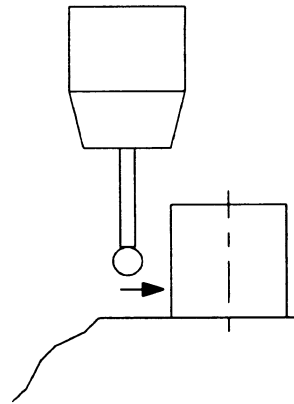
27. Measuring a full circle G46

Purpose

To measure a full circle and to determine the centre point coordinates and any deviation between the programmed circle radius and the calculated radius.



Measuring an inner circle



NB6902A

Measuring an outer circle

Format

Measuring an inner circle

N... G46 [Circle centre coords.] R... {1+1 J+1} {1+1 K+1} {J+1 K+1} {F...} {X1=...} N=... E...

Measuring an outer circle

N... G46 [Circle centre coords.] R... {1-1 J-1} {1-1 K-1} {J-1 K-1} {F...} {X1=...} N=... E...

Parameters

Circle parameters

X,Y,Z Center point coordinate
C Measurement target angle
P Point definition number
R Circle radius

Measuring parameters

I Measurement direction for X axis
J Measurement direction for Y axis
K Measurement direction for Z axis
F Feed between measurements
X1= Measurement path length

Measuring results

E Parameter-nr. measured radius
N= Point-nr.measured centre point

For absolute or incremental programming

X90=,Y90=,Z90= Absolute centre point
A90=,B90=,C90= Abs. measurement target angle
X91=,Y91=,Z91= Incremental centre point
A91=,B91=,C91= Incr. measurement target angle

Measuring a full circle G46

Modal words

E...=

Associated functions

G45, G49, G50

M26, M27, M28

Type of function

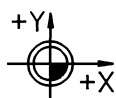
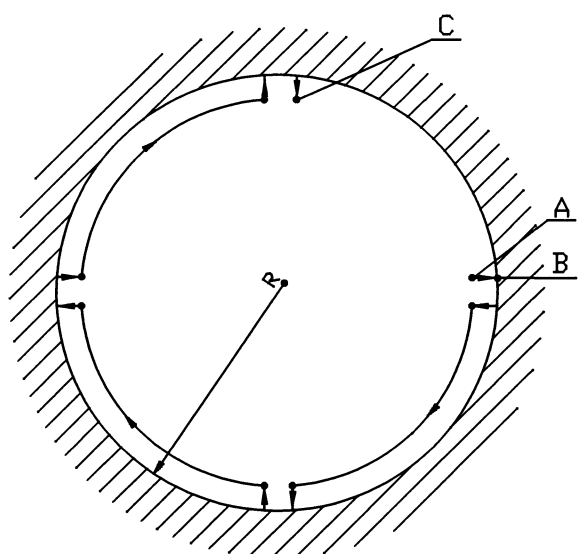
Non-modal.

Notes and Usage

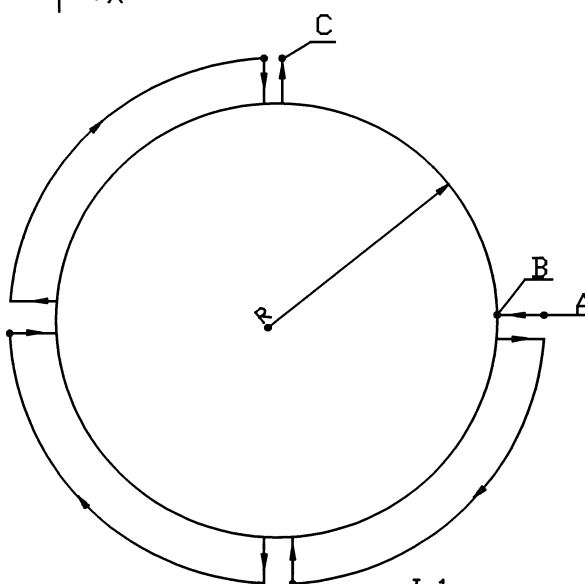
MEASURING THE POSITIONS

Four positions are measured when a G46 block is executed. The measurements take place as if four G45 blocks were programmed. So refer to G45 for additional information about PRE- AND POST- MEASURING DISTANCE, TOOL MEMORY, AIR BLOW and COLLISION PROTECTION.

MEASURING SEQUENCE



I+1
J+1



I-1
J-1

NB9890

A= Start point (is positioned with rapid speed)
B= Measurement point
C= End point

1. The probe moves rapidly to the pre-measuring position of the first point to be measured. This position is defined by the programmed circle centre, the programmed radius and the pre-measuring distance (X1=). This movement is executed with the positioning logic of G0.
2. The probe moves at a fixed feed rate (MC 843) towards the first point on the programmed circle. The probe can pass the point, however, it must be triggered along the path between the pre-(MC 844) and post-measuring(MC 845) distance.
3. When the probe is correctly triggered, the measured position is automatically stored. Then the probe moves back (at the set feed rate) to the starting position and with the programmed feed rate (F-word) along the circle in a clockwise direction until it reaches the second pre-measuring position.
4. The procedure just given is repeated for the second, third and fourth position.
5. When the fourth position has been measured, the circle centre and radius are calculated from the four measured points. The coordinates of the circle centre are stored in the Point Memory and the radius in the E- parameter memory.

MEASURING INNER OR OUTER CIRCLE (I/J/K)

Any pair of the addresses I, J, K simultaneously define the type of circle to be measured and the plane in which the circle is located. A pair of addresses must be stated in each G46-block.

Plane	Inner Circle		Outer Circle	
XY (G17)	1+1	J+1	1-1	J-1
XZ (G18)	1+1	K+1	I-1	K-1
XZ (G19)	J+1	K+1	J-1	K-1

STORING CENTRE POINT COORDINATES (N=)

The word N= states the number in the Point Memory where the calculated coordinates of the centre point are stored. Eg. N=12, means that the centre point coordinates are stored in P12.

STORING THE CIRCLE RADIUS (E)

The E-word states the number of the E-parameter where the calculated radius is stored. Eg. E45 means that the circle radius is stored as the value of E-parameter 45.

ERROR MESSAGES

An error message is displayed and the movement stops,

1. if the probe touches an obstruction during the movement to the premeasuring position,
2. if the probe exceeds the post-measuring distance.

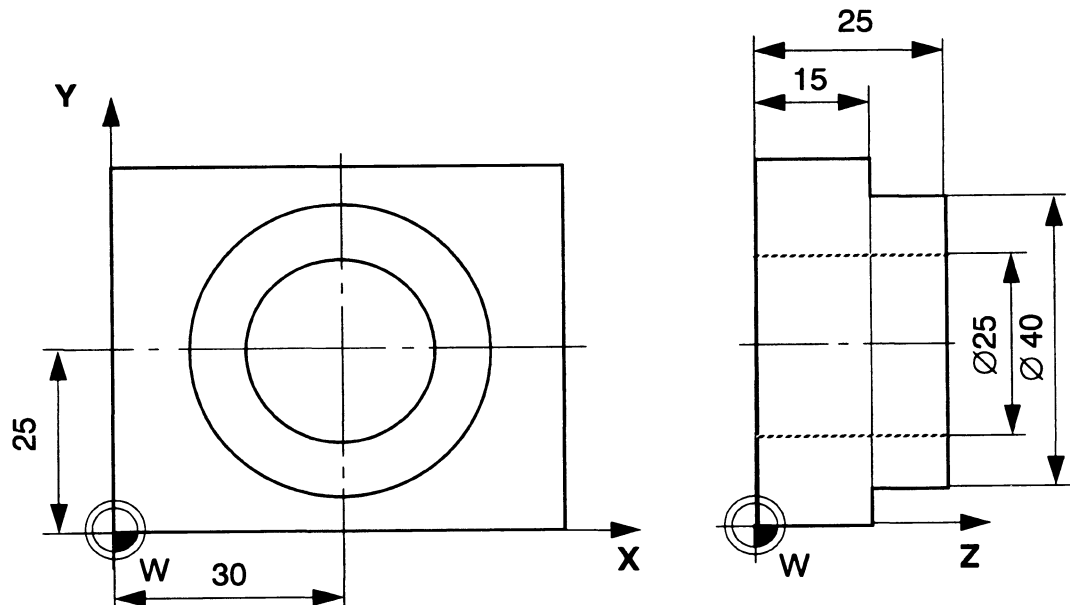
RESTRICTION

Only one circle can be measured by each G46-block.

note: The G46 function is also used in conjunction with the M26 function for probe calibration. Refer to G46 + M26 section for additional information.

Example

Measuring an inner and outer circle in the XY-plane



Measuring the inner circle:

N... G46 X30 Y25 Z20 I+1 J+1 R12.5 F3000 N=59 E24

Measuring the outer circle:

N... G46 X30 Y25 Z20 I-1 J-1 R20 F3000 N=58 E23

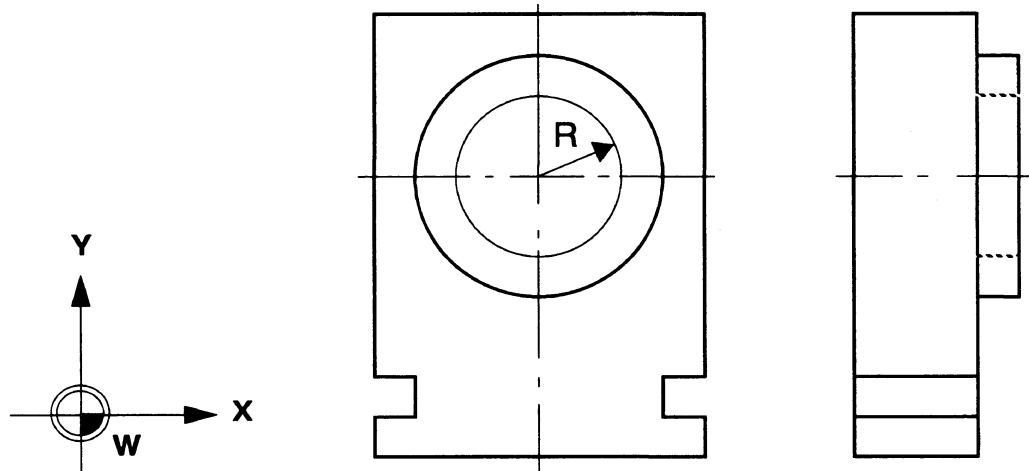
Explanation:

The circles are measured, the centre points of the measured circles calculated and stored in the Point Memory and the radius calculated and stored in the Parameter Memory.

28. Probe calibration G46+ M26

Purpose

To determine the radius of a touch trigger probe by touching a calibration ring, thus a ring gauge whose diameter is exactly known.



A probe must be calibrated:

- when the probe is used for the first time
- when a new stylus is used
- after any suspected bending of the stylus.

Note: It is assumed that the position of the ball centre relative to the spindle axis is already determined.

Format

Measuring an inner ring gauge

N... G46 {I+1 J+1} {I+1 K+1} {J+1 K+1} {F...} {X1=...} M26

Measuring an outer ring gauge

N... G46 {I-1 J-1} {I-1 K-1} {J-1 K-1} {F...} {X1=...} M26

Parameters

- I Measurement direction for X axis
- J Measurement direction for Y axis
- K Measurement direction for Z axis
- F Feed between measurements
- X1= Measurement path length

Modal words

E...=

Associated functions

G45, G46, G49, G50

M25, M27, M28

Type of function
non-modal

Notes and Usage

POSITION OF THE CALIBRATION RING

The position of the calibration ring and its radius are stored in the Machine Constant Memory.

MEASURING SEQUENCE

The measurements are executed in the same way as with G46. In stead of programming the position of the ring gauge, its coordinates are picked up from the Machine Constant Memory.

MEASURING INNER OR OUTER RING GAUGE (I/J/K)

Any pair of the addresses I, J, K simultaneously define the type of ring to be measured and the plane in which the ring is located. A pair of addresses must be stated in each G46-block.

Plane	Inner ring		Outer ring	
XY (G17)	I+1	J+1	I-1	J-1
XZ (G18)	I+1	K+1	I-1	K-1
XZ (G19)	J+1	K+1	J-1	K-1

UPDATING THE RADIUS OF THE PROBE

The difference between the radius of the ring stored in the Machine Constants and the measured radius is used to update the probe radius and store this value in the tool memory for the active tool (= the probe).

WHEN PROBE CALIBRATION IS REQUIRED

A probe should be calibrated in the cases mentioned above and also:

- if accuracy demands it.
- if the repeatability of relocation of the probe in the spindle is poor. In this case calibrating may be required each time the probe is selected.

Example

```
N46002
N1 G17
N2 T1 M6
N3 D207 M19
N4 G46 I1 J1 M26
N5 Z200 M30
```

Explanation:

- N1: Set the plane of operation to be the XY-plane
- N2: Load the touch trigger probe
- N3: Stop the spindle in a defined position
- N4: Calibrate the probe by moving it to the inside surface of a ring gauge located in the XY- plane. The measured radius of the touch probe is stored in the tool memory location of the active tool (T1). A default MC value is used for the pre-measuring distance.

29. Checking on tolerances G49

Note

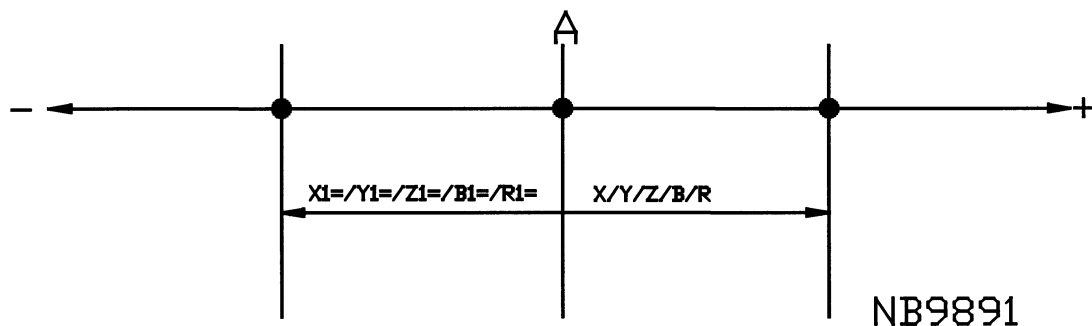
Use of this function is limited only to programs made on earlier control systems.

Since the G45 uses an internal memory, the G49 function can only be used together with G45. Basic measurement movement G145 uses E-parameters and may have the same functionality. It is therefore wiser to use the new basic measurement movement G145.

Purpose

To check whether the difference between a programmed value and the measured (G45/G46) value lies within set tolerance limits. If the difference is within the limits the program is allowed to continue. However, if the difference is not within limits there can be:

- a repeat of a section of the program until the difference is acceptable
- a conditional jump in the program
- a display of an error message.



A= Programmed point

The measured point must lie within the highest tolerance limit (X/Y/Z/B/R) and the lowest tolerance limit (X1=/Y1=/Z1=/B1=/R1=).

Format

A repeat of a program section

N... G49 {X...,X1=...} {Y...,Y1=...} {Z...,Z1=...} {B...,B1=...} {R...,R1=...} N1=... {N2=...} {E...}

The plane for the rotary table is determined by the definition of the 4th axis in the machine constant list. (MC117 should be 4 and MC118 should be B(66) or C(67)). B,B1 relates to the 4th axis B or C. R applies to the plane of the rotary table. Rotary axis A is not allowed.

A conditional jump.

N... G49 {X...,X1=...} {Y...,Y1=...} {Z...,Z1=...} {B...,B1=...} {R...,R1=...} N=... E...

Display an error message

N... G49 {X...,X1=...} {Y...,Y1=...} {Z...,Z1=...} {B...,B1=...} {R...,R1=...}

Parameters

Tolerance values

X Positive tolerance value in X
X1= Negative tolerance value in X
Y Positive tolerance value in Y
Y1= Negative tolerance value in Y
Z Positive tolerance value in Z
Z1= Negative tolerance value in Z
B Positive tolerance value in B
B1= Negative tolerance value in B
R Positive tolerance circle radius
R1= Negative tolerance circle radius

Conditional jump

E Jump condition: $E > 0$
N= Jump to blocknumber

Repeat of program section

N1= Repeater begin block
N2= Repeater end block

Modal words

E...=

Note

If the difference between the measured value and the programmed value is within the set tolerances, the program continues with the block after the G49 or G50.

Associated functions

G45, G46, G50

Type of function

Non-modal.

Notes and Usage

CONTINUATION IF VALUES ARE WITHIN LIMITS

If the difference between the measured value and the programmed value is within the set tolerances, the program continues with the block after the G49.

REPEAT OF A PROGRAM SECTION (E, N1=, N2=)

The words E, N1= and N2= are used to repeat a section of the program when a tolerance limit is exceeded.

The E-word specifies the number of repeats ($E > 0$).

When no number of repeats is programmed (no E-word is present), the sequence is repeated only once.

BLOCK NUMBERS OF REPEAT SEQUENCE (N1=, N2=) '

These block numbers must be in the same partprogram or subprogram.

If N2= is not programmed, only the block indicated by N1= is repeated the specified number of times.

ORDER OF BLOCKS TO BE REPEATED

The order of executing the blocks in the repeat sequence must be the same as the order originally programmed. So in the program block N1=.. must be before block N2=..

CONTINUATION AFTER THE REPEAT

Once the repeats are executed, the program continues with the block after the G49.

CONDITIONAL JUMP (N=, E)

The words N= and E are used to specify a conditional jump when a tolerance value is exceeded.

The value of parameter E must be greater than zero before a jump can occur.

The word N= states the block number in the same program or subprogram to which control will jump when E>0.

CONTINUATION IF NO JUMP IS EXECUTED

If no jump is executed, because the limits are not exceeded or E<=0, program execution continues with the block after G49.

ERROR MESSAGE

An error message is generated by the CNC, if the measured value exceeds a tolerance limit and neither a repeat of a program section nor a conditional jump is programmed.

CONTINUATION AFTER AN ERROR MESSAGE

After resetting the error program execution continues with the block after G49.

CHECKING ON THE HIGHEST AND LOWEST TOLERANCE LIMIT

If tolerance checks are used to see if a part is made within tolerances, two G49-blocks can be used. The order must be:

1. Check to see if the highest tolerance limit is exceeded. If this occurs the part is too big, so a jump out of the measuring section of the program is necessary.
2. Check to see if the lowest tolerance limit is exceeded. If this occurs the part is too small and has to be milled again, so a repeat of the milling section with an updated tool radius is necessary.

Examples

EXAMPLE 1: A repeat of a program section

N97 G49 X0.005 X1=0.002 N1=80 N2=95 E2

Explanation:

If the measured position is more than 0.005 mm higher or 0.002 mm lower than the programmed position, the program section from block number N80 to N95 is repeated two times. After the repeat program execution continues from the block after N97.

Alternative:

E2 is the desired position

E3 is measured position due to G145.

N97 G29 E0 E0=E3<(E2-0.002) N=100

N98 G29 E0 E0=E3>(E2+0.005) N=100

N99 G14 N1=80 N2=95 J2

N100 ...

Explanation:

N97 : Jump to N100 when measure position smaller is then 0.002 mm.

N98 : Jump to N100 when measure position greater is then 0.005 mm.

N99 : Repeat program twice.

EXAMPLE 2: A conditional jump

N197 G49 X0.005 X1 =0.002 E10 N=80

Explanation:

If the measured position is more than 0.005 mm higher or 0.002 mm lower than the programmed position and the value of parameter E10 is greater than zero, a jump in the program to block N80 is performed and program execution continues from that block.

Alternative:

E2 is the desired position.

E3 is the measured position due to G145.

N96 G29 E10 N=99

N97 G29 E0 E0=E3<(E2-0.002) N=80

N98 G29 E0 E0=E3>(E2+0.005) N=80

N99 ...

EXAMPLE 3: A conditional jump and repeat of program section

```
N10 G49 R.02 R1=2 E1 N=13 E1=1  
N11 G49 R2 R1=.02 N1=1 N2=6
```

Explanation:

- N10: If the measured position is more than 0.02 mm higher than the programmed position, jump to N13. R1= is set high to avoid that this limit is exceeded.
- N11: If the measured position is more than 0.02 mm lower than the programmed position, repeat the program section from N1 to N6. R is set high to avoid that this limit is exceeded.

Alternative:

E2 is the desired position of the circle radius.

E3 is measured position of the circle radius

```
N10 G29 E0 E0=E3>(E2+0.02) N=80
```

```
N98 G29 E0 E0=E3>(E2-0.02) N=95
```

```
N99 G14 N1=1 N2=16
```


30. Processing measuring results G50

Note

Use of this function is limited only to programs made on earlier control systems.

Since the G45 uses an internal memory, the G50 function can only be used together with G45. Basic measurement movement G145 uses E-parameters and may have the same functionality. It is therefore wiser to use the new basic measurement movement G145.

Purpose

To make corrections derived from the measured differences on either the zero offsets or the tool dimensions.

Format

To change zero offsets

With standard zero offsets or MC84=0:

N... G50 {X1} {I...} {Y1} {J...} {Z1} {K...} [{B1}{C1}{C2}] [{B1=..}{C1=..}] {L...} N=...

With MC84>0 zero offsets extends:

N... G50 {X1} {I...} {Y1} {J...} {Z1} {K...} [{B1}{C1}{C2}] [{B1=..}{C1=..}] {L...} **N=54.[nr]**

The plane for the rotary table is determined by the definition of the 4th axis in the machine constant list. (MC117 or MC120 should be 4 and the associated axes should be B(66) or C(67)). B1 or C1 relates to the 4th axis B or C. Rotary axis A is not allowed.

To change the tool length

N... G50 T... L1=1 {I...} {J...} {K...} {T2=...}

To change the tool radius.

N... G50 T... R1=1 {X1=...} {T2=...}

Parameters

Zero offsets

N= Offset-nr for correction (52-59)
 X X1: zero point shift in X
 Y Y1: zero point shift in Y
 Z Z1: zero point shift in Z
 B B1: zero point shift in B
 C C1: zero point shift in C
 I Multiplication factor for X
 J Multiplication factor for Y
 K Multiplication factor for Z
 L Multipl. factor for rotary-axis
 B1= Prog.angle in B after calculation
 C1= Prog.angle in C after calculation

Tool dimensions

T Tool dimensions to be corrected
 X1= Multiply factor for tool radius
 L1= L1=1: correction of tool length
 R1= R1=1: correction of tool radius

Modal words
E...=, T2=

Associated functions
G45, G46, G49

Type of function
Non-modal.

Notes and Usage

TO CHANGE OFFSET VALUES (N=)

With the G50 function new offset values derived from the measured corrections can be stored in the Zero Offset Memory.

MULTIPLICATION FACTOR FOR AXES (I,J,K,L)

A multiplication factor can be applied to the measured difference, eg. K.8, means multiply Z-axis difference by 0.8.

The multiplication factor can have a positive or negative value.

If no factor is stated, the default value +1 will be used automatically by the CNC.

MACHINE CONFIGURATIONS (B1,C1,C2)

B-Axis B1: Aligning of a work piece, mounted on a round table (B-axis) turning around the Y-axis, the measurement of two point in X and Z direction are sufficient. The rotations angle is in respect with the X-axis.

The round table is in the XZ-plane.

The work piece is turning around the Y-axis.

The measuring tracer stands in the Z-direction.

C-Axis C1: Aligning of a work piece, mounted on a round table (C-axis) turning around the Z-axis, the measurement of two point in X and Y direction are sufficient. The rotations angle is in respect with the X-axis.

The round table is in the XY-plane.

The work piece is turning around the Z-axis.

The measuring tracer stands in the Z-direction.

C-Axis C2: This is for special machine configurations, an extended possibility of C1. In this situation the C-axis is turned 90 degrees and rotates the C-axis around the X-axis, instead of round the Z-axis. Alignment of a work piece, mounted on a round table (C-axis) turning around the X-axis, the measurement of two point in Y and Z direction are sufficient. The rotations angle is in respect with the Y-axis.

The round table is in the YZ-plane.

The work piece is turning around the X-axis.

The measuring tracer stands in the Z-direction.

ALIGNING A WORK PIECE ON A ROTARY TABLE (B1=,C1=)

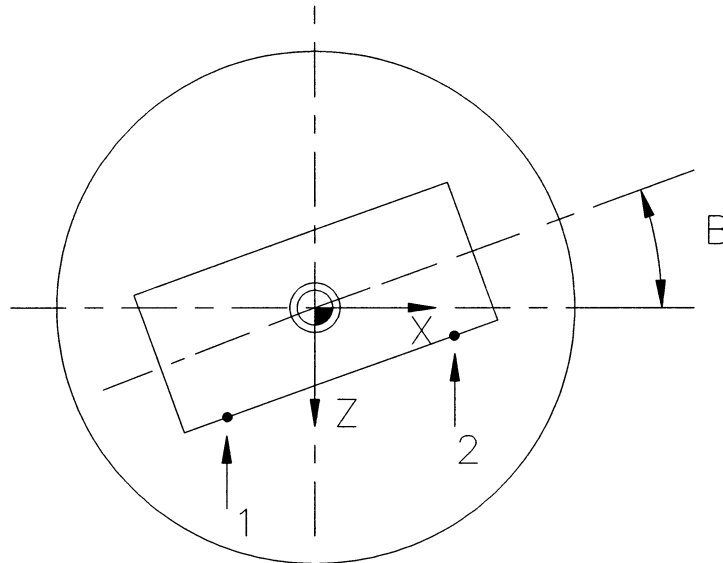
If a work piece is mounted on a table which rotates around the Y- or Z-axis, it is possible to align the work piece by measuring two points in the X- and Z- or X- and Z- direction.

The angle the work piece makes with the X-axis or Y-axis, is automatically calculated by the control and can be used to rotate the table, so that the work piece is parallel to the X-axis or Y-axis.

If the work piece makes initially an angle with the X-axis, this angle can be programmed with the word B1= or C1=.

If B1= or C1= is not programmed, B1=0 or C1=0 is assumed.

Refer to example 3. for the method how to use this feature.



NB9803

TO CHANGE TOOL DIMENSIONS (T)

With the G50 function new tool dimensions derived from the measured corrections can be stored in the Tool Memory.

MULTIPLICATION FACTOR FOR TOOL DIMENSIONS (I,J,K,X1=)

The multiplication factor for the tool radius is X1=.

The multiplication factor for the length correction depends on the active main plane defined by G17, G18 or G19:

K...	for Z-difference	(G17-plane is active)
J...	for Y-difference	(G18-plane is active)
I..	for X-difference	(G19-plane is active)

The multiplication factor can have a positive or negative value.

If no factor is stated, the default value +1 will be used automatically by the CNC.

Examples

EXAMPLE 1. Changing a stored zero offset

N... G50 X1 I0.8 N=54

Change the X-coordinate of the G54 offset by multiplying the correction by 0.8 and storing the new G54 X-coordinate value into the offset memory.

EXAMPLE 2. Changing a tool dimension

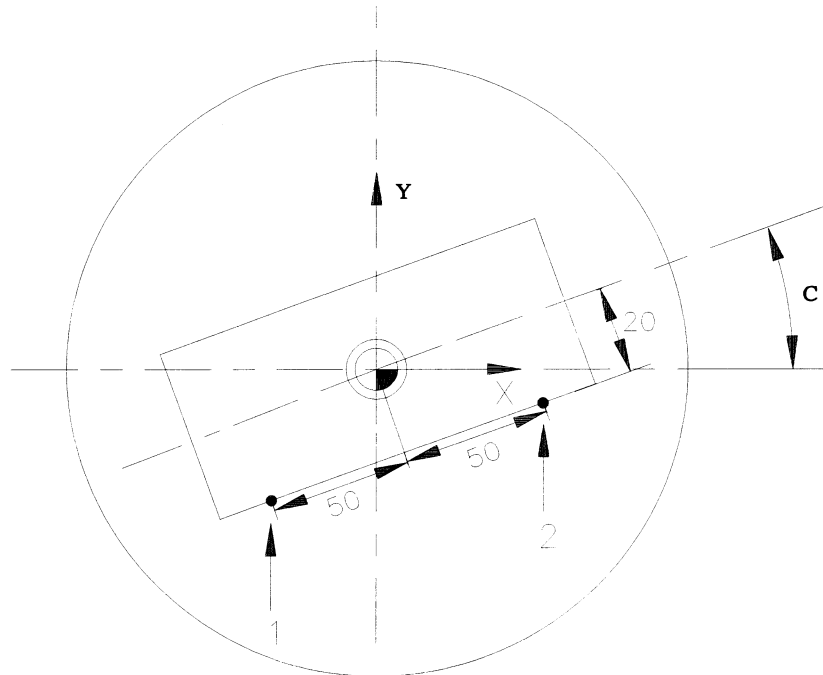
N... G50 T5 L1=1 K0.97 R1=1

Change the length of tool 5 by multiplying the Z-difference (tool in Z-axis) by 0.97, and store the new dimension into the tool memory.

EXAMPLE 3. Aligning a workpiece mounted on a rotary table

A part is mounted on a rotary table and should be aligned parallel to the X-axis. With a touch trigger probe two points on the part are measured and then the table is rotated over the calculated angle.

The Controller knows, when G45 is activated twice and G50 once that a rotatory table has to be measured.



NR9804

```

N 50003
N1 G17
N2 G54
N3 T1 M6
N4 M27
N5 G45 X-50 Y-20 Z0 C0 J1
N6 G45 X50 Y-20 Z0 J1
N7 G50 C1 N=54
N8 M28
N9 G54
N10 G0 Z100 C0

```

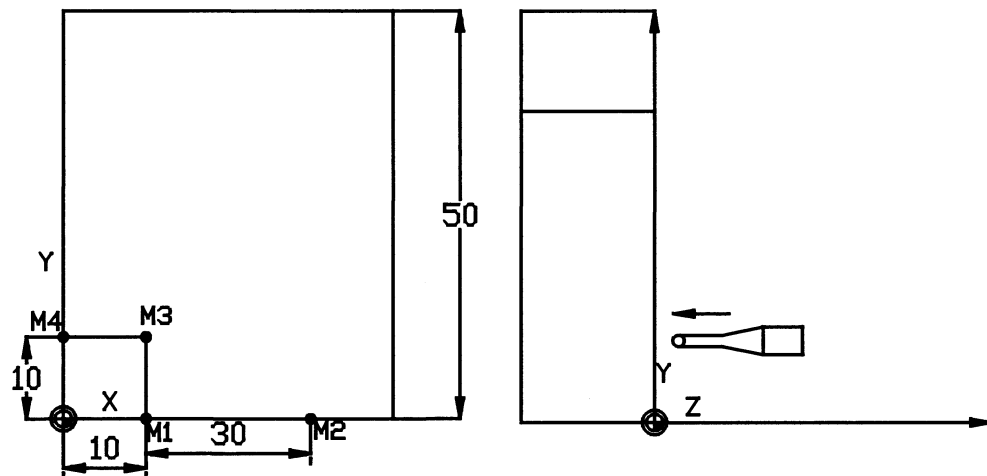
Explanation

N1: Set the plane of operation
 N2: Set the zero point
 N3: Load the touch trigger probe
 N4: Activate probe
 N5: Measure point 1
 N6: Measure point 2
 N7: Update of the zero offset value of the C-axis with the calculated angle
 N8: Deactivate probe
 N9: Set the zero point
 N10: Retract the tool and rotate the table to C0.

Note: N7 G50 C1 N=54 C1=30

If in block N7 C1=30, then the table rotates 30 Grad extra so that the table is parallel to the X-Axis.

EXAMPLE 4. Determining the zero point



The probe is standing in the Z-axis. The part is mounted on a table rotating around the Z-axis. Five points of the part (M1 to M4 and again M1) are measured. M1 and M2 for covering the angular displacement; M3, M4 and M1 for measuring the positions of the axes.

The section of the part program for determination the zero point could be:

```

N50004
N1 G54
N2 G17
N3 G0 X10 Y-10 Z10 T1 M6
N4 M27
N5 G45 X10 Y0 Z-5 C0 J1
N6 G0 Z10
N7 G45 X40 Y0 Z-5 J1
N8 G0 Z10
N9 G50 C0 N=54
N10 G54
N11 G0 C0
N12 G45 X10 Y10 Z0 K-1
N13 G0 Z10
N14 G45 X0 Y10 Z-5 I1
N15 G0 Z10
N16 G45 X10 Y0 Z-5 J1
N17 G0 Z50
N18 G50 X1 Y1 Z1 N=54
N19 G54
N20 M28

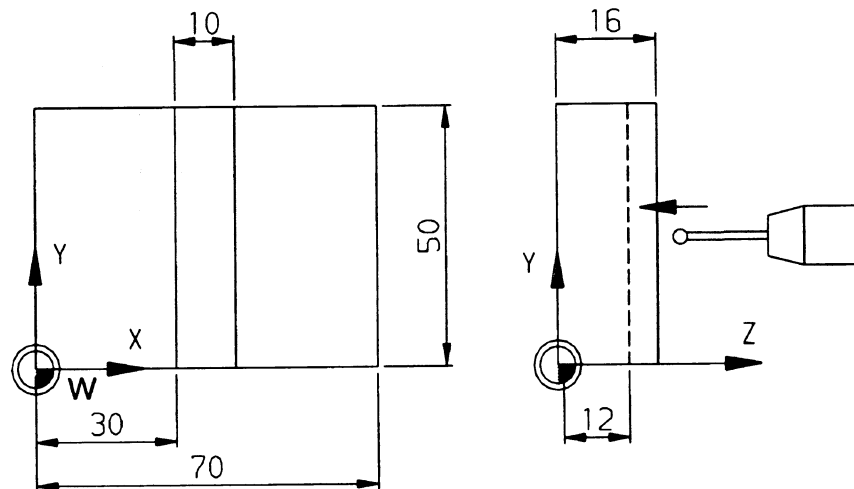
```

Explanation

N1: Set the zero point
 N2: Set the plane of operation to be the XZ-plane
 N3: Load the touch trigger probe and move to the programmed position
 N4: Activate probe
 N5: Measure point M1
 N6: Retract the probe to avoid collision
 N7: Measure point M2
 N8: Retract the probe to avoid collision

N9: Update of the zero offset value of the C-axis with the calculated angle
 N10: Set the zero point
 N11: Rotate the table to C0.
 N12: Measure point M3 to determine the position in the tool axis
 N13: Retract the probe to avoid collision
 N14: Measure point M4 to determine the position in the X-axis
 N15: Retract the probe to avoid collision
 N16: Measure point M1 to determine the position in the Z-axis
 N17: Retract the probe to avoid collision
 N18: Update of the zero offset values of the X-, Y- and Z-axis
 N19: Set the updated zero point
 N20: Deactivate probe

EXAMPLE 5. Correcting the length of a tool



With a mill a groove is made, the depth of the groove is measured and the tool length of the mill updated.

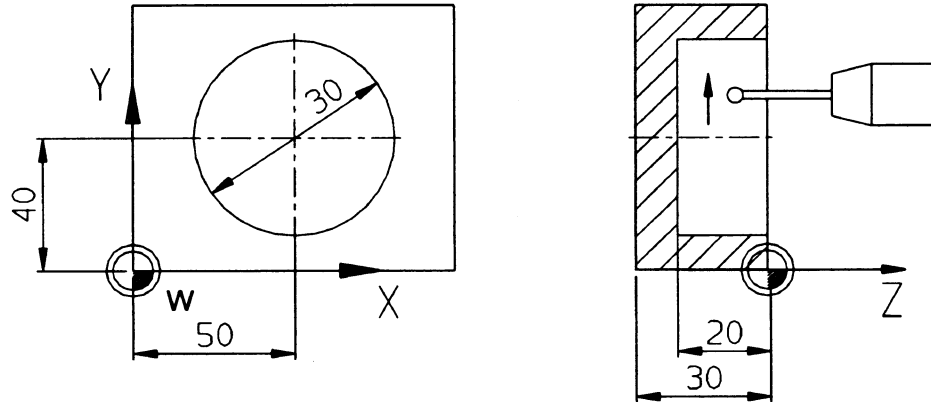
N90005
 N1 G17
 N2 T1 M6 (Mill radius 5 mm)
 N3 X35 Y60 Z12 S1000 M3
 N4 G1 Y-10 F200
 N5 G0 Z200 M5
 N6 T2 M6 (Probe)
 N7 M27
 N8 G45 X35 Y25 Z12 K-1
 N9 G50 T1 L1=1
 N10 M28
 N11 Z200 M30

Explanation:

N1: Set the plane of operation to be the XY-plane
 N2: Load the mill of 10 mm diameter
 N3: Start the spindle and move the mill to the start point of the groove
 N4: Mill the groove
 N5: Retract the tool and stop the spindle
 N6: Load the probe
 N7: Activate probe

- N8: Measure the point in the negative direction of the tool axis
 N9: The calculated difference in the Z-axis is used to correct the length of tool 1
 N10: Deactivate probe
 N11: Retract the probe and end of program

EXAMPLE 6. Milling and measuring a hole



A hole is milled and measured with a touch trigger probe. Checks are provided to see if the tolerance on the radius of the hole is within the required limits. If the radius is too small, the hole is milled again. If the radius is too large the part is rejected and an message displayed.

The part program could be:

```

N50006
N1 G54
N2 G17
N3 T1 M6 (Mill radius 5 mm)
N4 G89 Z-20 B2 R15 K6 F300 S1000 M3
N5 G79 X50 Y40 Z0
N6 G0 Z50 M5
N7 T2 M6 (probe)
N8 M19
N9 M27
N10 G46 X50 Y40 Z-10 R15 I1 J1 F500 E5
N11 G0 Z50
N12 G49 R.02 R1=2 N=15 E5 (Hole > (15+.02) jump to N=19)
N13 G49 R2 R1=.02 N=18 E5 (Hole < (15-.02) jump to N=15)
N14 G29 E1 E1=1 N=21 (jump to the end of program)
N15 G50 T1 R1=1
N16 M28
N17 G14 N1=3 N2=6
N18 G29 E1 E1=1 N=21
N19 MO (HOLE OUT OF TOLERANCE)
N20 M30
  
```

Explanation:

- N1: Set the zero point
 N2: Set the plane of operation to be the XY-plane
 N3: Load a mill with a diameter of 10 mm
 N4: Define the fixed cycle for milling the hole

N5: Mill the hole
N6: Retract the tool and stop the spindle
N7: Load the touch trigger probe
N8: Spindle stop at a certain angle
N9: Activate probe
N10: Measure the hole at four points
N11: Retract the probe to avoid collision
N12: Check to see if the radius of the hole is not too big, (less than $15+.02$). If the radius is too big, reject the part and display a message to it.
N13: Check to see if the radius of the hole is not too small, (greater than $15-.02$). If the radius is too small, update the radius value in the tool memory and mill the hole again.
N14: Jump to the end of the program
N15: Update of tool radius in tool memory
N16: Deactivated probe
N17: Repeat of the blocks N3 to N6 to mill the hole within tolerance.
N18: Jump to the end of the program
N19: Stop the program execution and display a message
N20: End of program

31. Cancel/Activate G52 zero point shift G51/G52 (till V310)

Note

Use of these functions is limited only to programs made on earlier control systems.

Purpose

Fix Workpiece zero point with programmed value.

Format

To activate and re-activate after G51 has been used:

N... G52

To cancel:

N... G51

Notes and usage

The function G52 can be cancelled with softkey CLEAR CONTROL or overridden by programming G51.

The functions G51 and G52 will stay active after Softkey CANCEL PROGRAM, M30 or turning off the controller.

If there is a zero point shift G54.. G59 active. Then G52 is active with the zero offset. Is G52 active and G54.. G59 are activated then G54..G59 are activated with the zero offset of G52.

Cancel/Activate G52 zero point shift G51/G52 (till V310)

32. Cancel/Activate pallet zero point shift G51/G52 (from V310)

Purpose

Fix pallet zero offset with programmed value.

Format

Activate and deactivate after G51 has been used:

N... G52

To cancel:

N... G51

Notes and usage

The function G52 can be cancelled with softkey CLEAR CONTROL or overridden by programming G51 .

The functions G51 and G52 will stay active after Softkey CANCEL PROGRAM, M30 or turning off the controller.

If there is a zero point shift G54.[nr] active. Then G52 is active with the zero offset. Is G52 active and G54.[nr] are activated then G54.[nr] are activated with de zero offset of G52.

Function G52 is used for the purpose of automation, for instance pallet control. In this case the values for G52 are set by an IPLC program.

From V320

When **MC84 = 0** then G52 is stored in ZO.ZO (zero point).

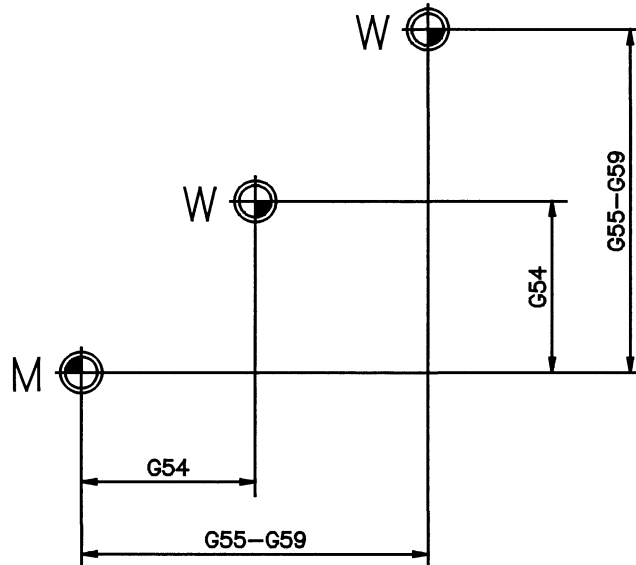
When **MC84 > 0** then G52 is stored in PO.PO (Pallet Offset).

In ZO.ZO and in PO.PO the zero points can be edited.

33. Cancel/Activate zero point shift G53/G54 - G59 (MC84=0)

Purpose

To shift the workpiece zero point to a new position whose coordinate values are stored in the zero point memory (under the appropriate number).



- G53 Cancels the active G54-G59 zero point
- G54 Activates G54 zero point shift
- :
- G59 Activates G54 zero point shift

Format

To activate:
N... [G54/G55/G56/G57/G58/G59]

To cancel:
N... G53

Parameters

M Machine function

Associated functions

G51/G52, G92, G93

Type of function

Modal

Notes and Usage

MACHINE ZERO POINTS

If a machine has several clamping stations or more than one rotary table it is necessary to state secondary machine zero points. These points are related to the geometric machine zero point (M_0). The axial distances measured from M_0 specify the position of these secondary zero points and are stored in the Zero Offset Memory together with their identifying G-function.

THE ZERO OFFSET MEMORY

All values for the G54-G59 zero point shifts (zero offset values) must be stored in the appropriate memory by using the user's panel or a data carrier before the program is executed.

G52 ZERO POINT SHIFT

The function G52 is not influenced by one of the functions G53 to G59.

ABS./INCR. ZERO POINT SHIFTS G92/G93

A programmed zero point shift (G92 or G93) is cancelled by one of the functions G53 to G59.

SCALING, MIRROR IMAGE AND AXIS ROTATION (G73, G92/G93)

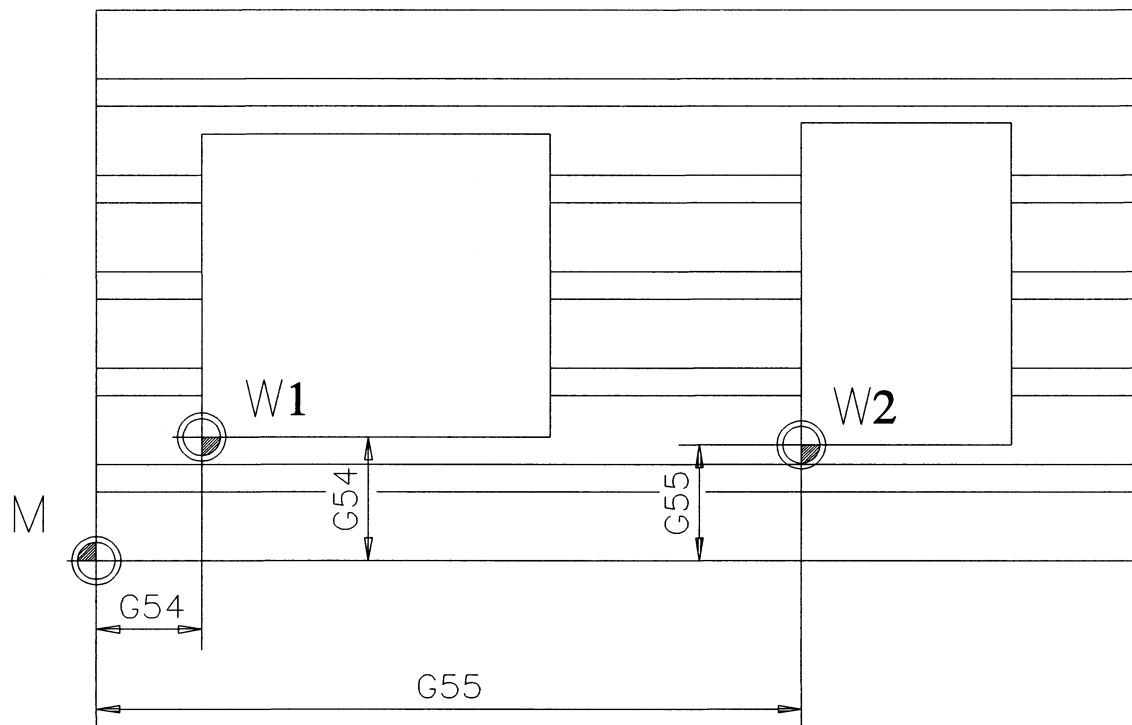
One of the functions G53 to G59 can be used in a program sequence which is scaled, mirrored or rotated. The zero point shift is performed in the coordinate system of the machine tool and not influenced by the programmed change of coordinates.

CANCELLATION

A secondary machine zero point can be overridden by programming G53.

G53 is automatically set at switching on the control and after a reference point search.

The functions G54 to G59 are not cancelled by CLEAR CONTROL, M30 or softkey CANCEL PROGRAM.

Example

```

N60  G54
:
N600  G55
:
N700  G53

```

Explanation:

N60: Select the secondary machine zero point W1. Its coordinates (X40, Y100,Z300) are retrieved from the Zero Offset Memory.

All programmed coordinates are measured from W1.

N600: Select the secondary machine zero point W2. Its coordinates (X200, Y100,Z100) are retrieved from the Zero Offset Memory.

Machine zero point W1 is cancelled and W2 is now active, therefore all programmed coordinates are measured from W2.

N700 : Cancel with G53 machine zero point W2. Then the coordinates (X0,Y0,Z0) are cleaned.

Zero point W2 will be deleted and M(machine zero point) will be active. After this action all coordinates are measured from M.

Cancel/Activate zero point shift G53/G54 - G59 (MC84=0)

34. Zero point shift extension G54 MC84>0 (from V320)

Apart from the present zero point shift table G54..G59 there is another zero point shift table G54 I[nr] showing a maximum of 100 zero point shifts. The appropriate zero point shift is selected by machine constant MC84.

The functionality is the same as that of the present zero point shift memory G54..G59, except for the following extensions and differences:

- 100 potential zero point shifts in the zero point shift memory
- Identification of zero point memory Ze.Ze (MC84 > 0)
- Programming (shift values) of zero point shift in NC program
- Programming of angle of rotation (B4=) in zero point shift
- Programming of zero point shift with an index (G54 I[nr])
- Comment is entered in zero point shift memory

Purpose

To shift the workpiece zero point to a new position. The coordinate values can be entered in the zero point shift memory or programmed in the NC program block.

Format

Define and use zero point shift as follow:

G54 I[nr] [Axis coordinates] {B4=..}

Use zero point shift as follow:

G54 I[nr]

Parameters

X,Y,Z	Zero point coordinate
M	Machine function
A	Zero point angle
B	Zero point angle
I	Zero point index

A zero point shift can contain up to 6 Axis coordinates.

Associated functions

G50, G51, G52, G53, G54 ... G59, G92, G93, G149, G150

Type of function

Modal

Notes and usage

NUMBER OF ZERO POINTS

The number of possible zero point shifts in the table is determined by a machine constant (MC84). (0< MC84 <99).

CHANGING MACHINE CONSTANT MC84

The zero point shift table is adjusted in the event of scaling (MC84 > 0). The existing zero points are maintained. Extended zero points are initialized to zero.

Attention:

If MC84 is zeroed, the table is changed (ZE.ZE is changed to ZO.ZO). The new zero point table is initialized to zero.

ENTERING IN ZERO POINT SHIFT MEMORY

Shift values can be entered in the zero point memory in two different ways:

- The values of zero point shifts G54 I[nr] are entered into the zero point shift memory via the control panel or through a data carrier before the program is executed.
- The values of zero point shift G54 I[nr] X.. Y.. Z.. A.. B.. C.. B4=.. are programmed in an NC program block. When the program is edited, the programmed values are accepted in the zero point shift memory and activated.

Attention:

If no new zero point shift values have been programmed in the program block, the zero point shift values already stored in the memory are not overwritten or deleted. The axis coordinates not programmed are taken from the memory. Risk of collision!

COMMENT

Additionally, each zero point shift in the table may be commented.

AXIS ROTATION

Additionally, each zero point shift in the table may involve an axis rotation.

First, the shift is executed and then the coordinate system is rotated through angle B4=.

MACHINE ZERO POINTS

If a machine has several clamping stations or more than one rotary table it is necessary to state secondary machine zero points. These points are related to the geometric machine zero point (M_0). The axial distances measured from M_0 specify the position of these secondary zero points and are stored in the Zero Offset Memory together with their identifying G-function.

ZERO POINT SHIFT MEMORY

All values of zero point shifts G54 I[nr] should be stored in the zero point shift memory via the control panel or by a data carrier, before the program is executed.

G52 ZERO POINT SHIFT

G52 does not affect the functions G53...G59. If G52 is active, G54..G59 will be effective from this shift onwards.

ABSOLUTE/INCREMENTAL ZERO POINT SHIFTS (G92/G93)

A programmed zero point shift (G92 or G93) is deleted from any of the G54 I[nr] functions.

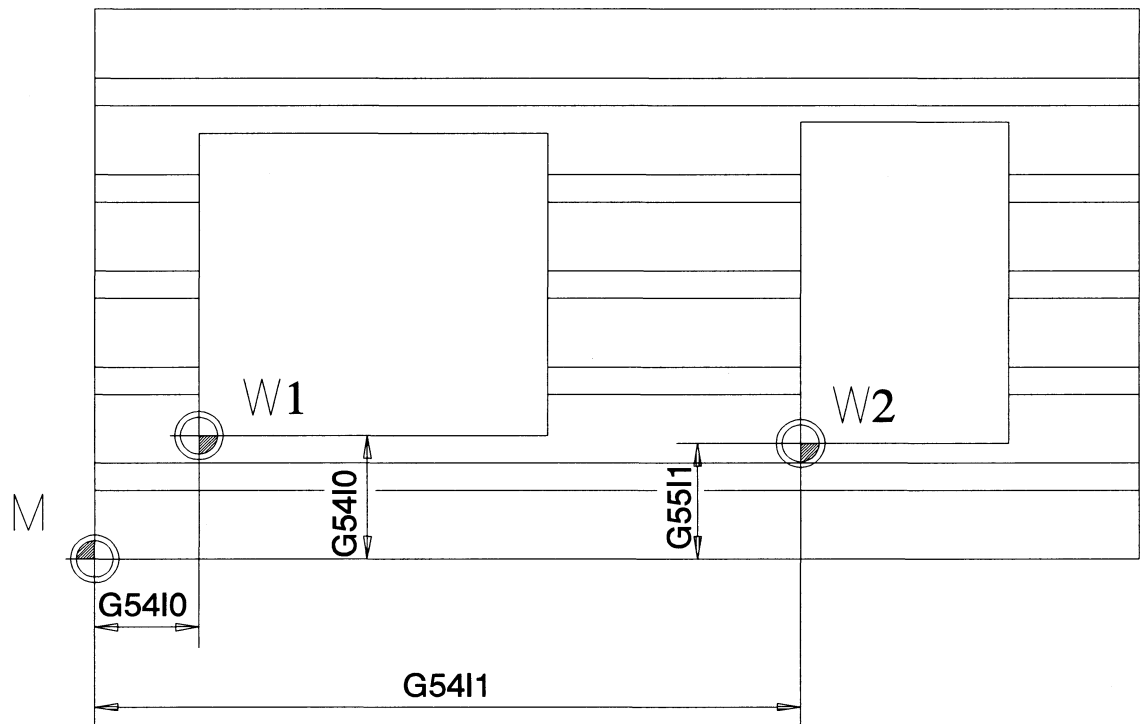
SCALING, MIRRORING AND AXIS ROTATION (G73, G92/G93)

It is allowed to use any of the functions G54 I[nr] in a program section which is to be scaled, mirrored or rotated. The zero point shift takes place in the coordinate system of the machine tool and is not affected by the programmed change of coordinates.

DELETE

G54 I[nr] is automatically deleted by the CLEAR CONTROL softkey and by programming G53.

The functions G54...G59 are not deleted when the CANCEL PROGRAM softkey or M30 is used.

Example

```

N60 G54 I1
:
N600 G54 I2
:
N700 G53

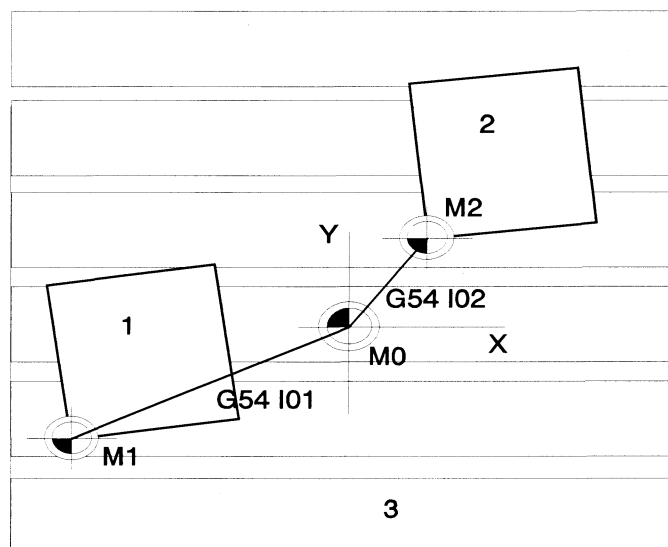
```

Explanation:

- N60 : Selection of zero point W1. The coordinates (X40,Y100,Z300) are taken from the zero point shift memory. All programmed coordinates are measured from W1.
- N600 : Selection of zero point W2. The coordinates (X200,Y100,Z100) are taken from the zero point shift memory. Zero point W1 is deleted and W2 activated. All programmed coordinates are then measured from W2.
- N700 : Disable zero point W2. The coordinates (X0,Y0,Z0) are taken from the G53 zero point shift memory. Zero point W2 is deleted and M activated. All programmed coordinates are then measured from M.

Example

Axis rotation



1= Workpiece 1

2= Workpiece 2

3= Table

Entry in zero point table and calling:

N60 G54 I1 X-42 Y-15 B4=14 (Z0 C0)

Zero point shift values are entered in the zero point shift table.

Machine workpiece 1, all programmed coordinates are measured from M1.

N120 G54 I2 X10 Y24 B4=-17

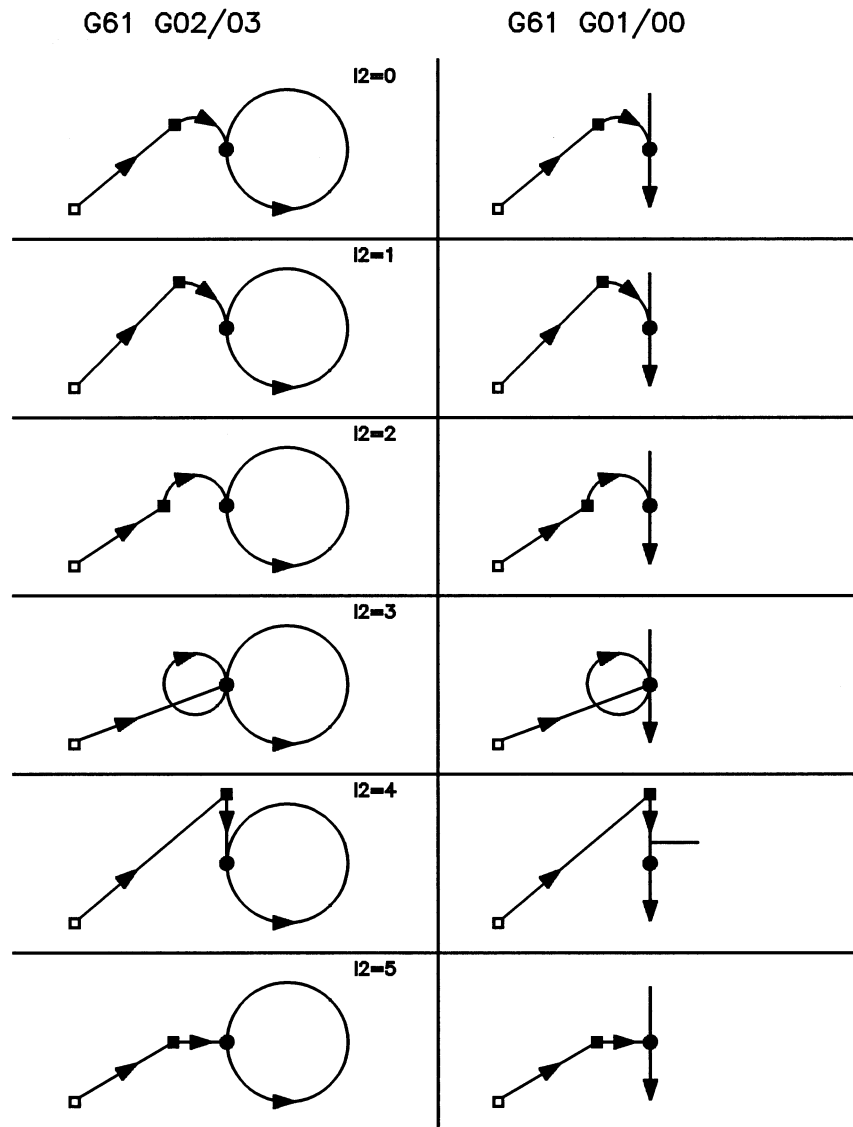
Machine workpiece 2, all programmed coordinates are measured from M2.

35. Tangential approach G61

Purpose

Programs a tangential approach movement between a starting point and start of a contour.

Format



TANGENTIAL APPROACH MOVEMENT TO THE CONTOUR G61

- Actual position.
- Calculated start point in mainplane. Z1 can be programmed. If Z1 is not programmed then Z1=Z.
- Start point contour (X, Y, Z).

N... G61 {I2=} X... Y... Z... R... [{X1=} {Y1=} {Z1=}] {I1=} {F2=}

N... G61 {I2=} B2=... L2=... Z... R... [{X1=} {Y1=} {Z1=}] {I1=} {F2=}

Parameter

X,Y,Z Endpoint tangential approach
 P Point definition number
 R Radius
 Z1= Startpoint in Z
 X1= Startpoint in X
 Y1= Startpoint in Y
 B2= Polar angle
 L2= Polar length
 I1= Linear movement 0=rapid,1=feed
 I2= Tangential approach definition
 F2= In feed

I2=0 with Lines and semicircles
 I2=1 with quarter circle
 I2=2 with semicircle
 I2=3 Helix for feeding (for pockets)
 I2=4 Parallel to contour
 I2=5 Vertical

For absolute or incremental programming
 X90=,Y90=,Z90= Absolute endpoint
 X91=,Y91=,Z91= Incremental endpoint

Modal words
 F,S,T,T1=,T2=,M,H,E..=
 S1=,M1=

Associated functions

G0, G1, G2, G3
 Wordwise absolute/incremental programming (X90=..., X91=..)

Type of function

Not modal

Notes and usage

STARTPOINT OF THE APPROACH MOVEMENT

The Controller calculates the Startpoint. The first movement is a positioning movement towards the calculated startpoint. From this position the approach movement towards the contour is started.

APPROACH MOVEMENT

The approach movement consists of two movements. The first movement is a rapid or a feed movement(depends if is I1=0 or I1=1) towards the calculated start point. The second movement is a feed movement to the starting point of the contour.

POSITIONING THE STARTING POINT IN TOOL-AXIS

The Z(G17) parameter contents the contour startpoint in the tool-axis. The parameter Z1(G17) contents the startpoint of the approach movement.

APPROACH SIDE

The parameters G41 or G42 determine the approach side. (G41 is left, G42 is Right). When G40 is active the approach is the same as the in G41(left side).

THE CIRCULAR ARC MOVEMENT

The circular movement is determined by the position of the starting point of the contour and the starting point of the tangential movement. The Direction of the tangential circular movement is determined by the direction of the contour movement. The movement is always fluid between the tangential circular arc and the contour movement.

RADIUS COMPENSATION

If the radius compensation (G41/G42) is activated right before G61 is active the compensation will be activated during the linear movement. The actual position determines if the circular movement is small or big.

If the radius compensation is active the linear movement and the circular movement are operated according to the radius compensation.

PERPENDICULAR APPROACH MOVEMENT

The position of the contour startpoint determines the position of the perpendicular approach movement.

G1-FUNCTION

If there is no G-function programmed after a G17 Block G1 will not be active

LIMITATION WHEN I2=0 IS ACTIVE

When the actual position is further away than one diameter from the circular movement then the approach movement contains a linear and a circular movement. When the actual position is within the circular movement then parameter I2= changes from 0 to 1, and the begin movement contains a quarter circle movement.

TANGENTIAL APPROACH

PROGRAM

```

N33 G17
N34 G0 X.. Y.. (S)
N35 G41
N36 G61 X.. Y.. R.. (B)
N37 G64

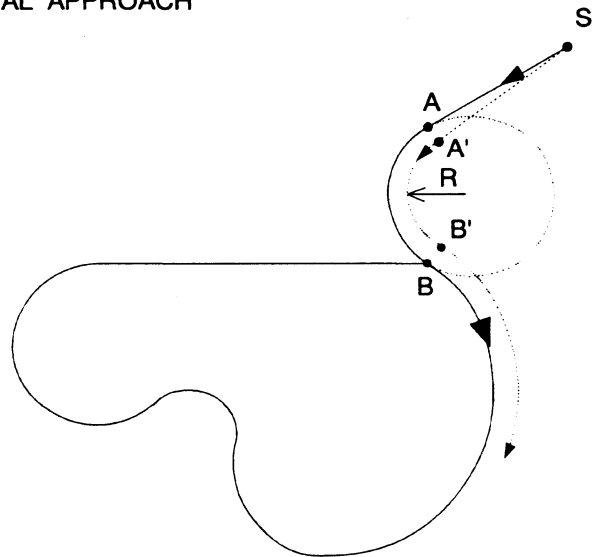
```

EXECUTION

```

N33 G17
N34 G0 X.. Y.. (S)
N35 G41
N36 G1 X.. Y.. (A')
N37 G3 X.. Y.. I.. J.. (B')
N38 G64

```



NB9866

Explanation:

- N33 : Define the main plane. All the movement occur in one plane.
 N34 : Moving to the startpoint position.
 N35 : Radius compensation must be active for the left side.
 N36 : Tangential approach movement (I2=0). This movement contents two movements one a linear and a circular movement.
 N37 : Start a contour.

LIMITATIONS

Programming a G61 has the following limitations:

- G61 is in G64 mode NOT allowed
- G61 is in Teach-In-Mode NOT allowed
- G61 is in Teach-In/Playback-Mode NOT allowed
- G61 is in G182-Mode NOT allowed

The programblock after G61 has limitation. The following functions are allowed:

- G64
- G0, G1, G2, G3 with movements in de active plane

Notes:

The programmer must program the approach movement in such away that during milling the contour doesn't damage.

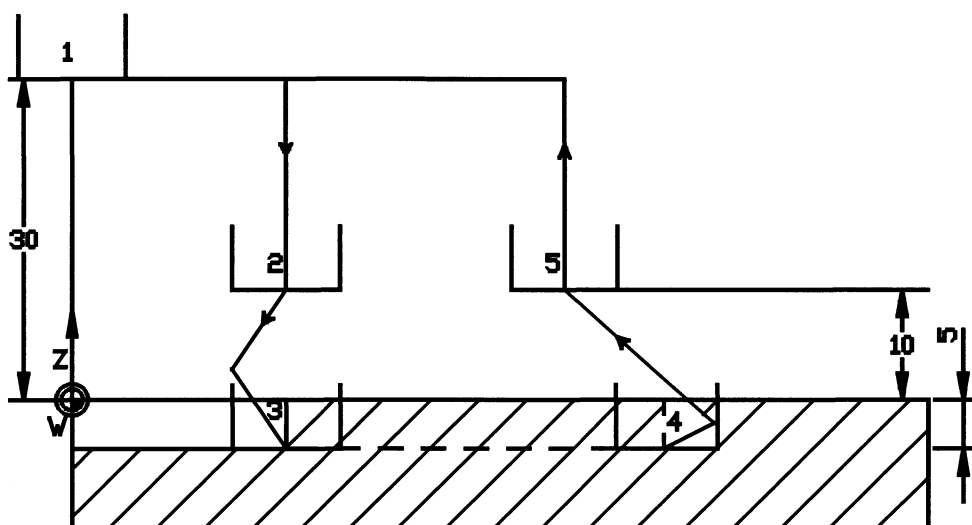
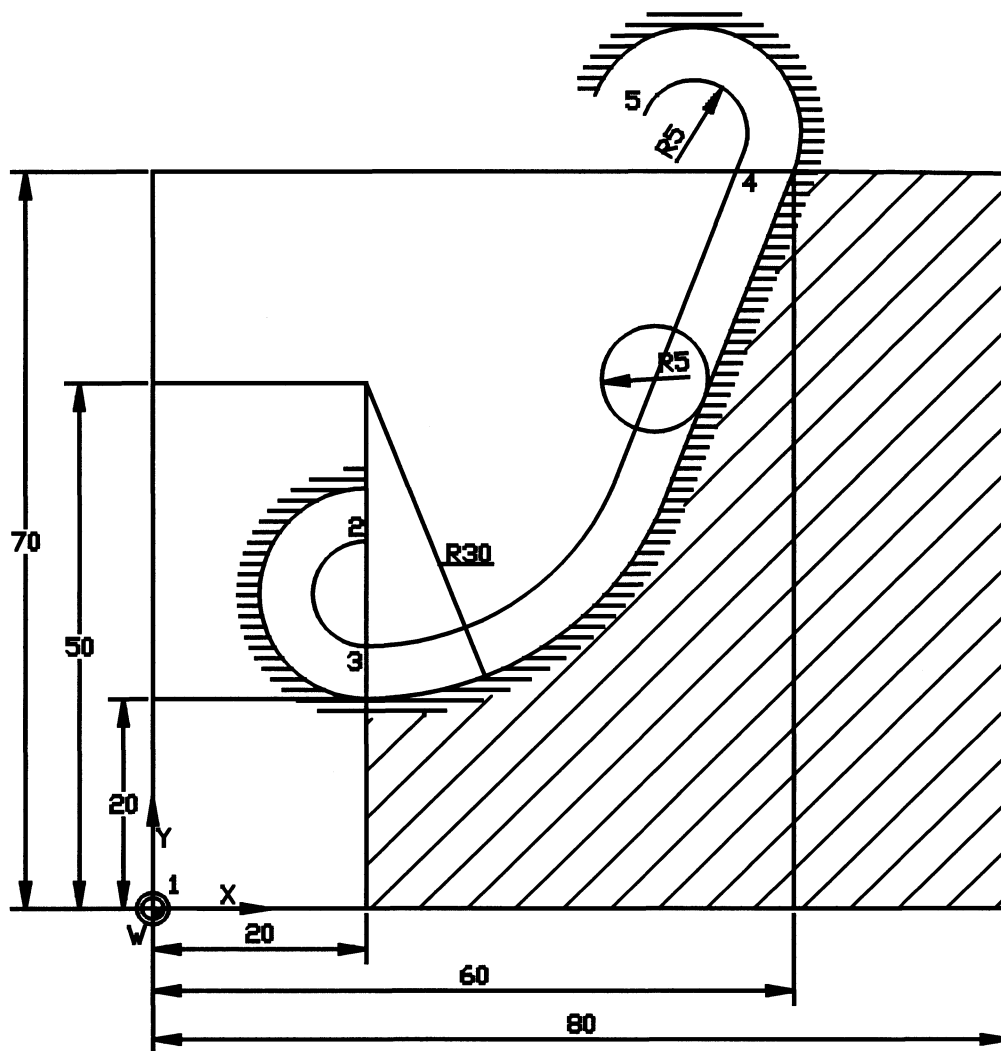
SUPPORT PROGRAMMING

The functions "tangential approach" (G61) and "tangential exit" (G62) can be used in the following mode:

- FREE ENTRY
- SUPPORT ENTRY

The SUPPORT ENTRY supports the programmer with pictures and text.

Example



```
N1 G17
N2 T1 M6 (Mill R5)
N3 F500 S1000 M3
N4 G0 X0 Y0 Z50
N5 G41
N6 G61 I2=2 X20 Y20 Z-5 Z1=10 R5 I1=0 F2=200
N7 G64
N8 G3 I20 J50 R1=0
N9 G1 X60 Y60
N10 G63
```

Explanation:

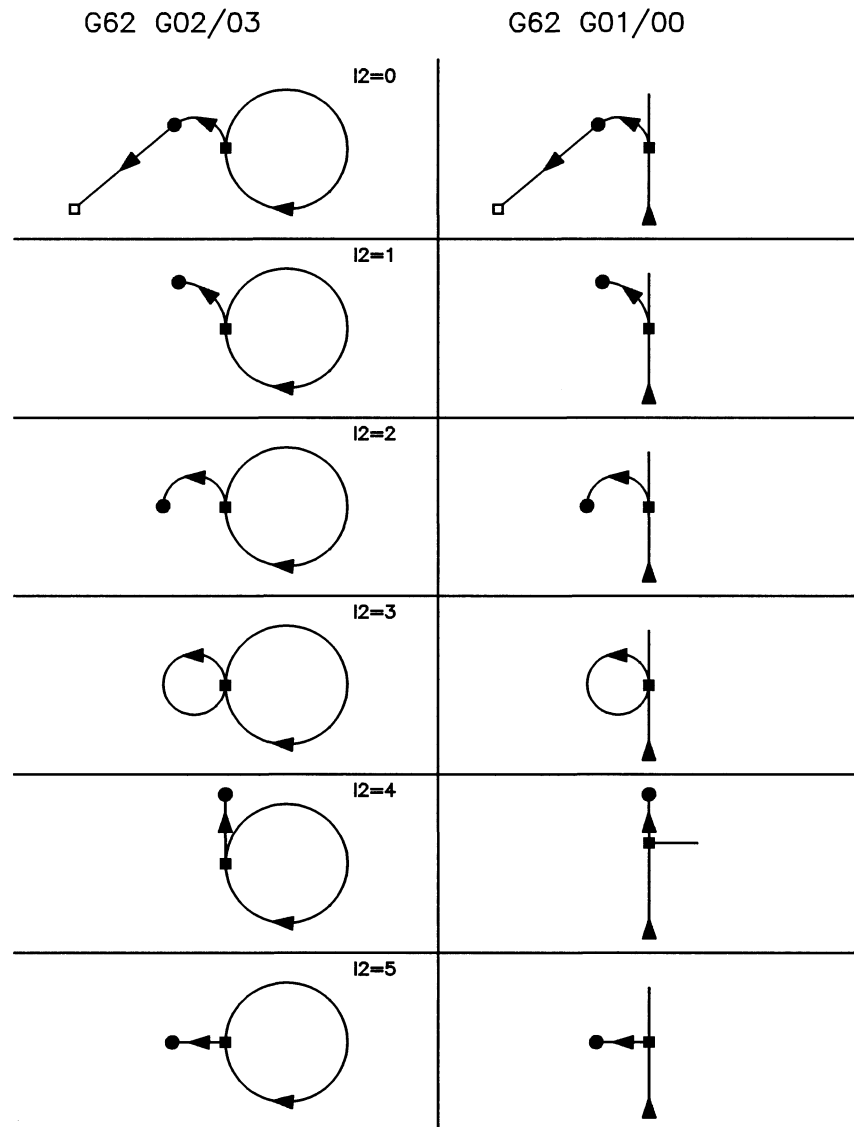
N1: Active XY-plane (G17).
N2: Load Tool T1.
N3: Activating feedrate, speed and spindle rotation(M3).
N4: Move tool rapidly to programmed position (position 1: X0 Y0 Z30).
N5: Set radius compensation LEFT
N6: -Tangential approach movement
-I2=2 is a semicircle
-The first section is a rapid movement with Positioning logic to the startpoint of the semicircle movement (position 2: X.. Y.. Z10).The Radius compensation is activated in this movement.
-The circular movement will preform a helix movement.
The contour starts with position X20 Y20 Z-5.(position 3: X20 Y25 Z-5).
N7: Start the contour description.
N8: A circular movement tangential to a Line.
N9: Tangential Linear movement.
N10: End of the contour description.

36. Tangential exit G62

Purpose

Programs a tangential exit after the end of the contour.

Format



TANGENTIAL EXIT FROM DE CONTOUR G62:

- End point contour.
- Calculated end point in mainplane. Z1 can be programmed. If Z1 is not programmed then the value doesn't change.
- Programmed end point tangential exit (X, Y, Z) (only I2=0).

N... G62 I2>0 Z1=... R... {I1=} {F2=}

N... G62 I2=0 X... Y... Z... Z1=... R... {I1=} {F2=}

N... G62 I2=0 B2=... L2=... Z... R... {I1=} {F2=}

Parameter

X,Y,Z Endpoint tangential exit
 P Point definition number
 R Radius
 X1= Startpoint in X
 Z1= Startpoint in Z
 Y1= Startpoint in Y
 B2= Polar angle
 L2= Polar length
 I1= Linear movement 0=rapid,1=feed
 I2= Tangential exit definition
 F2= In feed

I2=0 with Lines and semicircles
 I2=1 with quarter circle
 I2=2 with semicircle
 I2=3 Helix for feeding (for pockets)
 I2=4 Parallel to contour
 I2=5 Vertical

For absolute or incremental programming
 X90=,Y90=,Z90= Absolute endpoint
 X91=,Y91=,Z91= Incremental endpoint

Modal words
 F,S,T,T1=,T2=,M,H,E..=
 S1=,M1=

Associated functions

Wordwise absolute/incremental programming (X90=..., X91=...)

Type of Function

Not modal

Note:

To understand the function G62 read first function G61.

Notes and usage**CANCEL RADIUS COMPENSATION**

If the radius compensation (G40) is deactivated right before G61 is active the compensation will be deactivated during the tangential movement.

If the radius compensation is not deactivated the tangential movement and the linear movement operate according to the Radius compensation.

LIMITATIONS

Programming a G62 has the following limitations:

- G62 is in G64 mode NOT allowed
- G62 is in Teach-In-Mode NOT allowed
- G62 is in Teach-In/Playback-Mode NOT allowed
- G62 is in G182-Mode NOT allowed

The programblock after G61 has limitation. The following functions are allowed:

- G64
- G0, G1, G2, G3 with movements in de active plane

SUPPORT PROGRAMMING

The functions "tangential approach" (G61) and "tangential exit" (G62) can be used in the following mode:

- FREE ENTRY
- SUPPORT ENTRY

The SUPPORT ENTRY supports the programmer with pictures and text.

TANGENTIAL EXIT

PROGRAM

```

N51 G1 X.. Z.. (E)
N52 G63
N53 G40
N54 G62 X.. Z.. R.. (F)

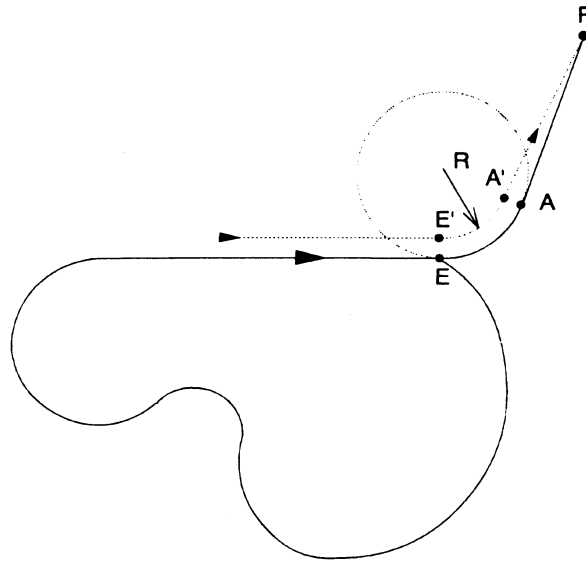
```

EXECUTION

```

N51 G1 X.. Z.. (E')
N52 G63
N53 G3 X.. Z.. R.. (A')
N54 G40
N55 G1 X.. Z.. (F)

```



NB9870

Explanation:

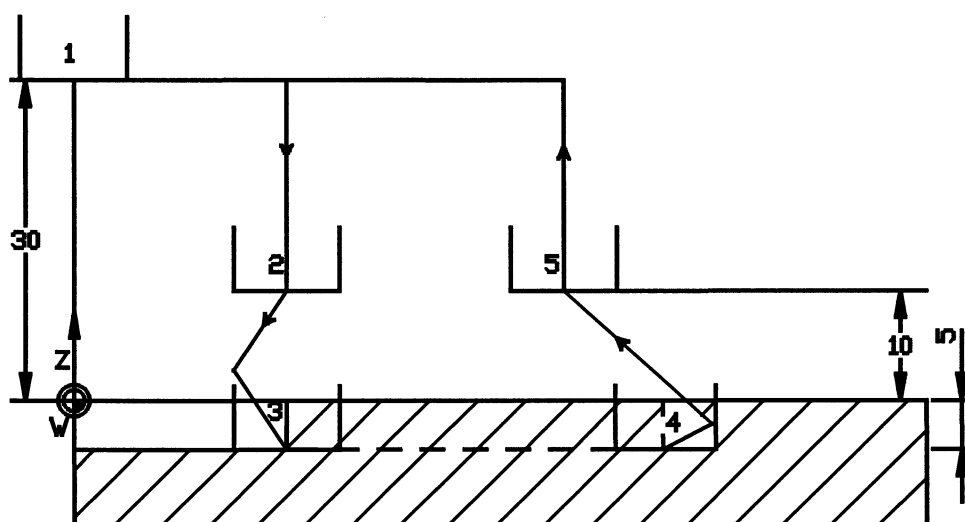
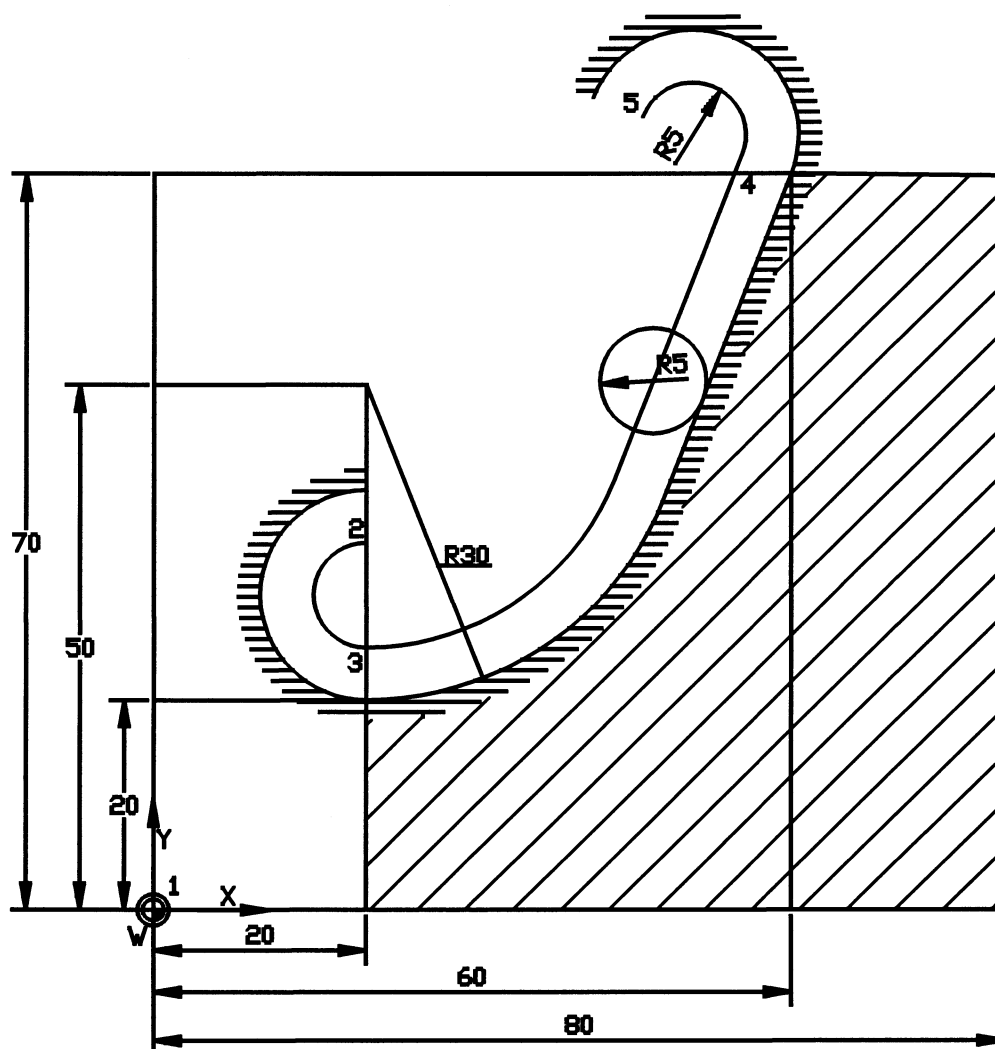
N51 : Last movement in the contour description.

N52 : End contour description.

N54 : Tangential Exit (I2=0). This movement contains two movements a circular and a linear movement.

N53 : The radius compensation is deactivated between the circular and the Linear movement.

Example




```

N1 G17
N2 T1 M6 (Mill R5)
N3 F500 S1000 M3
N4 G0 X0 Y0 Z50
N5 G41
N6 G61 I2=2 X20 Y20 Z-5 Z1=10 R5 I1=0 F2=200
N7 G64
N8 G3 I20 J50 R1=0
N9 G1 X60 Y60
N10 G63
N11 G62 I2=2 Z1=20 R5
N12 G40
N13 G0 X0 Y0 Z50
N14 M30

```

Explanation:

N1: Active XY-plane (G17).
 N2: Load Tool T1.
 N3: Activating feedrate, speed and spindle rotation (M3).
 N4: Move tool rapidly to programmed position (position 1: X0 Y0 Z30).
 N5: Set radius compensation LEFT
 N6: -Tangential approach movement
 -I2=2 is a semicircle
 -The first section is a rapid movement with Positioning logic to the startpoint of the semicircle movement (position 2: X.. Y.. Z10). The Radius compensation is activated in this movement.
 -The circular movement will preform a helix movement.
 The contour starts with position X20 Y20 Z-5.(position 3: X20 Y25 Z-5).
 N7: Start the contour description.

 N8: A circular movement tangential to a Line.
 N9: Tangential Linear movement with endpoint (position 4: x.. Y.. Z-5).
 N10: End of the contour description.
 N11: -Tangential Exit
 -I2=2 is a semicircle
 -The circular movement will preform a helix movement. Startpoint in the Z-axis is -5, and the endpoint is 10.
 N12: Radius compensation will be deactivated.
 N13: Retract Tool (position 1: X0 Y0 Z30)
 N14: Program End.

37. Cancel/Activate geometric calculations G63/G64

Purpose

G63: To cancel the geometric calculations and to return to programming complete blocks.

G64: To activate the geometric calculations.

General principles for using the geometry

Between the functions G64 and G63 a contour can be described. An easy way of programming linear and circular movements makes it possible to let the control perform the necessary calculations for e.g. an intersection point or point of tangency. Each time a calculation is required at least two blocks of data are used. Each block is programmed with the standard G-functions for linear (G0 and G1) and circular movements (G2 and G3) and some information to define the line or circle. These blocks do not necessarily contain all data as previously specified, but with some special words (indicators) is achieved, that the missing data can be calculated by the control.

The first block establishes where the start point is located and what type of end point is required. The second block supplies the data for calculating the end point coordinates of the first block as e.g. a point of tangency or an intersection point of two elements. This end point is also the start point of the second block.

Between these movements can be inserted:

- a chamfer (between linear movements),
- a rounding (between intersecting elements),
- a connecting circle (between tangent elements or elements which do not meet)

It may happen that the second block does not supply enough data for calculating the end point of the first block. In that case the control looks for the next block and try to calculate the end point of the second block and first one. Up to five blocks are looking for in advance.

Formats

Activating geometric calculations

N.. G64

Cancelling geometric calculations

N.. G63

Linear (G0/G1) and circular movements (G2/G3)

Only the most commonly used formats are given here. Refer to a special appendix at the end of this manual for a detailed description of the possible formats for G0/G1 and G2/G3 in the many cases and also for examples of the use of the geometry.

For all formats the G64-function is assumed to have already been programmed in a previous block and is therefore active.

The XY-plane is also assumed to be the active plane. Refer to Notes and usage PLANE SELECTION for changes to be made in the formats, if another plane is active.

Parameters

X,Y,Z Endpoint coordinate
X1=,Y1=,Z1= Arbitrary endpoint coordinate
P Point definition number
P1= Point definition number
I Center point in X / pitch in X
J Center point in Y / pitch in Y
K Center point in Z / pitch in Z
R Circle radius
B1= Angle
B2= Polar angle
L2= Polar length
B3= Polar angle for center
L3= Polar length for center
I1= Parallel shift
J1= 1=intersection left, 2=right
K1= Connection circle number
R1= R1=0 tangent to line

Modal words

F, F1=, S, T, H, E and M except M6, M66 and M67

Associated functions

None

Type of function

Modal

Notes and Usage

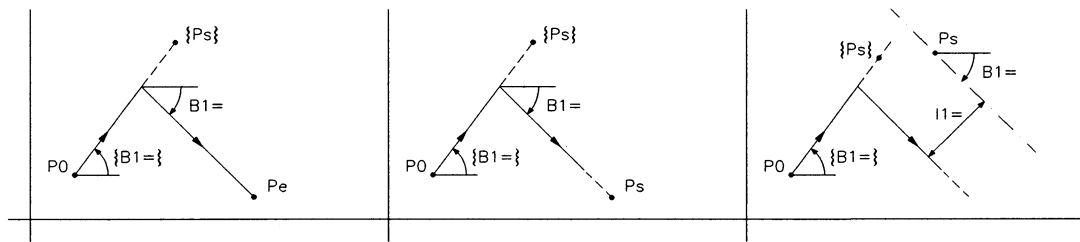
Within function G64 it is not possible to active G4.

Explanation of the possible formats:

In the illustrations in which the formats are explained, the following conventions are used:

P_0 = a start point known from the previous block
 P_s = a support point on a line or on a parallel line
 P_e = a programmed end point
M = a programmed circle centre point
R = a programmed radius of a circle

INTERSECTION POINT BETWEEN TWO STRAIGHT LINES



NB9665

Intersection point with known start point from first line

Possible definition of the first line

N.. G1 {B1=..}

(Startpoint and angle)

N.. G1 X{Ps} Y{Ps}

(Startpoint and support point)

Possible definition of the second line

N.. G1 B1=.. X{Pe} Y{Pe}

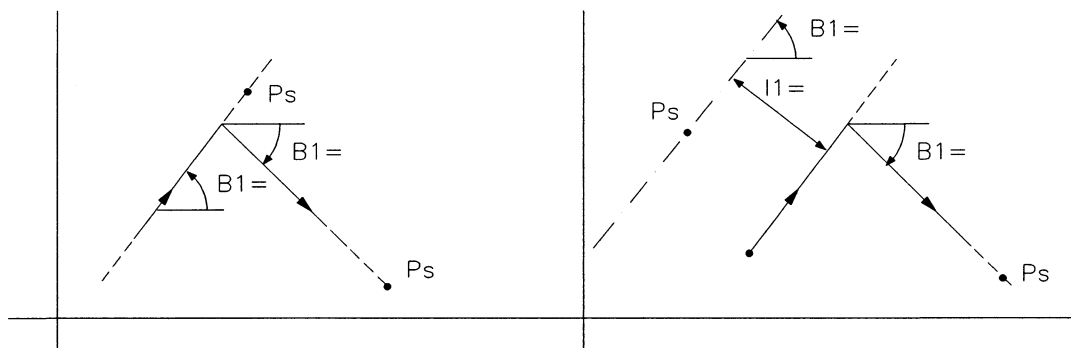
(Angle and end point)

N.. G1 B1=.. X{Ps} Y{Ps}

(Angle and support point)

N.. G1 X{Ps} Y{Ps} B1= I1=

(Support point, angle and parallel line)



NB9667

Intersection point with unknown start point from first line

Possible definition of the first line

N.. G1 {B1=..} X{Ps} Y{PS}

(Support point and angle)

N.. G1 B1= X{Ps} Y{Ps} I1=

(Support point, angle and parallel line)

Possible definition of the second line

N.. G1 B1=.. X{Pe} Y{Pe}

(Angle and end point)

N.. G1 B1=.. X{Ps} Y{Ps}

(Angle and support point)

N.. G1 X{Ps} Y{Ps} B1= I1=

(Support point, angle and parallel line)

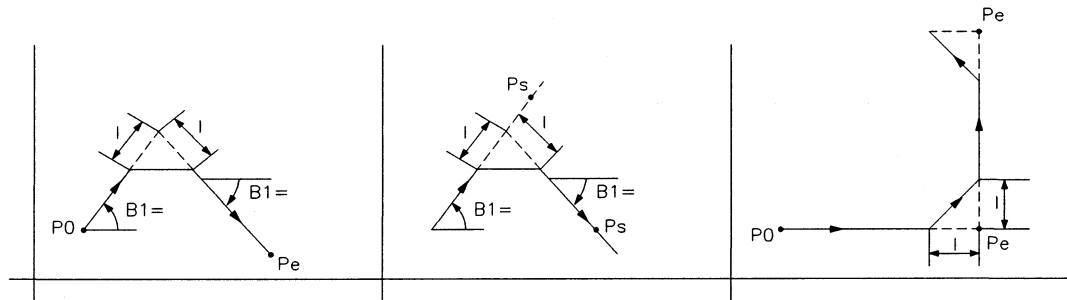
Note: It is also possible to program the intersection point as an end point, if it is known from the drawing. Refer to END POINT in Notes and usage for details.

A CHAMFER INSERTED BETWEEN TWO INTERSECTING LINES

The lines are programmed as indicated in the previous section.

The chamfer is:

- symmetrically located around the intersection point
- programmed with G1 and the length of the chamfer (I-word)



NB9682

Chamfer between two straight lines

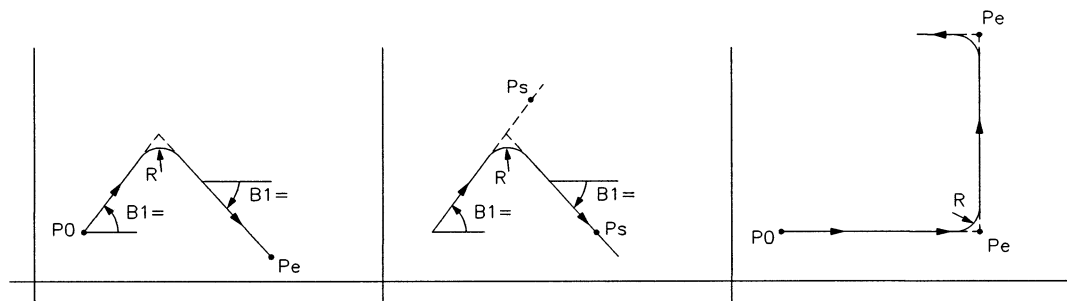
- N.. G1 {B1=..} or {support point/parallel line}
- N.. I.. {chamfer parameter}
- N.. B1=.. {end point} or {support point/parallel line}

A ROUNDING BETWEEN TWO INTERSECTING LINES

The lines are programmed as indicated in the previous section.

The rounding is:

- tangent to the line from the previous block and to the line of the next one.
- programmed with:
 - G2 or G3 indicating the direction of movement,
 - the radius (R-word) of the rounding.



NB9683

Rounding between two straight lines

- N.. G1 {B1=..} or {support point/parallel line}
- N.. G2/G3 R..
- N.. G1 B1=.. {end point} or {support point/parallel line}

Note: It is also possible to insert a rounding between a straight line and a chamfer or between a chamfer and a straight line.

N.. G1 {B1=..} or {support point/parallel line}
 N.. G2/G3 R..
 N.. I.. {chamfer parameter}
 N.. G2/G3 R..
 N.. G1 B1=.. {end point} or {support point/parallel line}

INTERSECTION POINT INDICATOR

If an intersection point between line and circle or two circles should be calculated, two points are possible.

With the word J1= is indicated which point is required:

- J1=1: the left intersection point (P1)
- J1=2: the right intersection point (P2).

With a line through the circle centre point

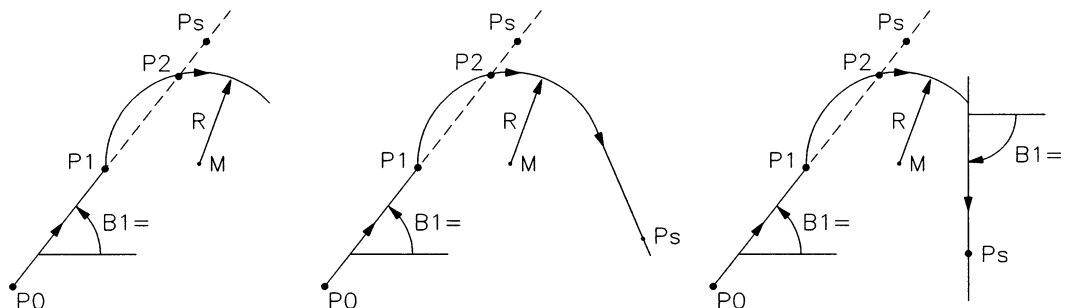
- J1=1 : the smallest distance between the start point or end point
- J1=2 : the largest distance between the start point or end point.

With a line that starts or ends in the circle centre.

- J1=1 : is the first intersection.
- J1=2 : is the second intersection.

(For additional information refer to Notes and usage INTERSECTION POINT INDICATOR)

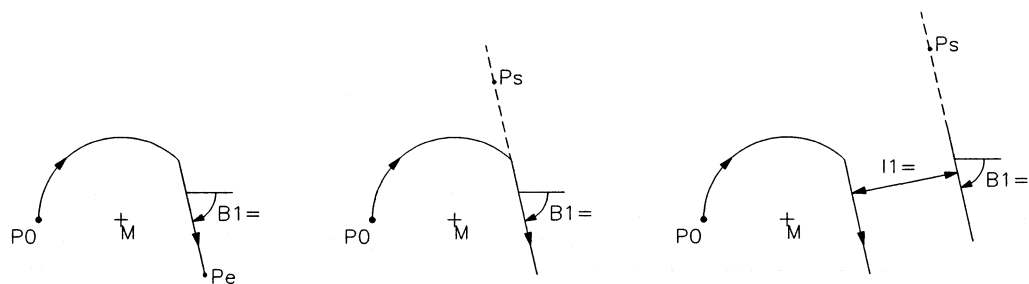
INTERSECTION POINT BETWEEN LINE AND CIRCLE OR CIRCLE AND LINE



NB9748

Line to circle

N.. G1 {B1=..} or {support point} J1=1/2
 N.. G2/G3 I.. J.. R..

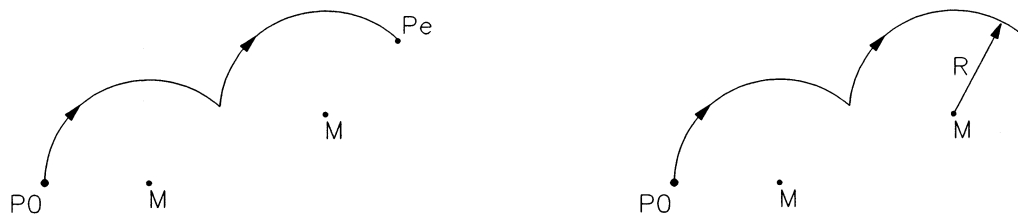


NB9692

Circle to line

N.. G2/G3 I.. J.. R.. J1=1/2
N.. G1 B1=.. {support point} or {end point}

INTERSECTION POINT BETWEEN TWO CIRCLES



NB9699

N.. G2/G3 I.. J.. {R..} J1=1/2
N.. G2/G3 I.. J.. R..

PROGRAMMING A ROUNDING

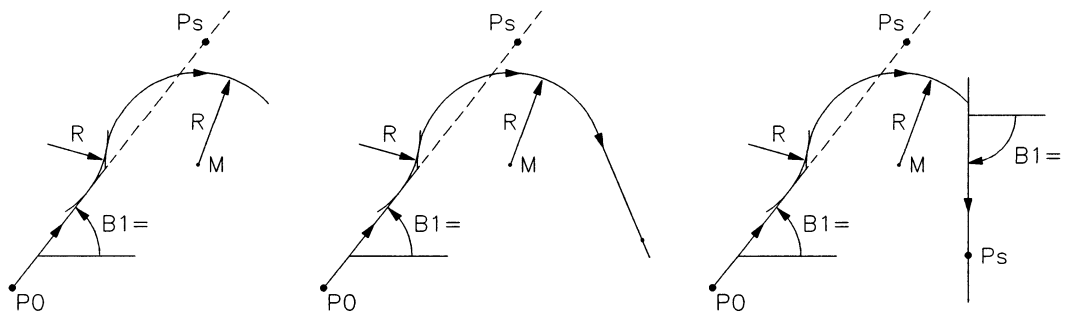
A rounding is:

-always tangent to the geometry elements (line or circle) from the previous block and the next one.

-programmed with:

-G2 or G3 indicating the direction of movement,
-the radius (R-word) of the rounding.

A ROUNDING BETWEEN INTERSECTING LINE - CIRCLE OR CIRCLE - LINE



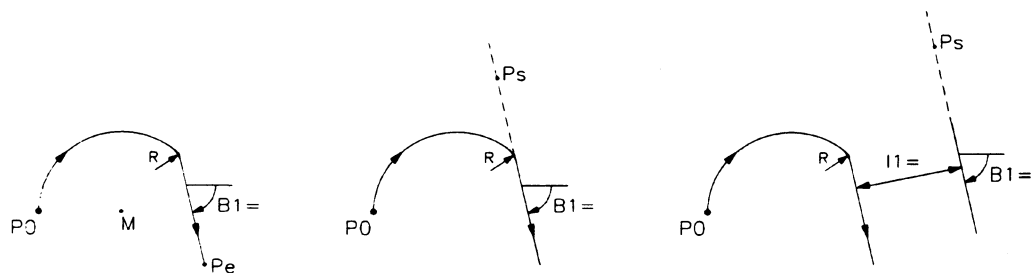
NB9690

Line to circle

N.. G1 {B1=..} or {supportpoint} J1=1/2

N.. G2/G3 R..

N.. G3/G2 I.. J... R..



NB9697

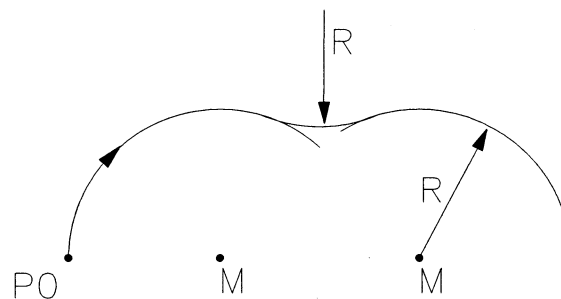
Circle to line

N.. G3/G2 I..... J... {R..} J1=1/2

N.. G2/G3 R..

N.. G1 B1=.. {supportpoint} or {endpoint}

A ROUNDING BETWEEN TWO INTERSECTING CIRCLES



NB9703

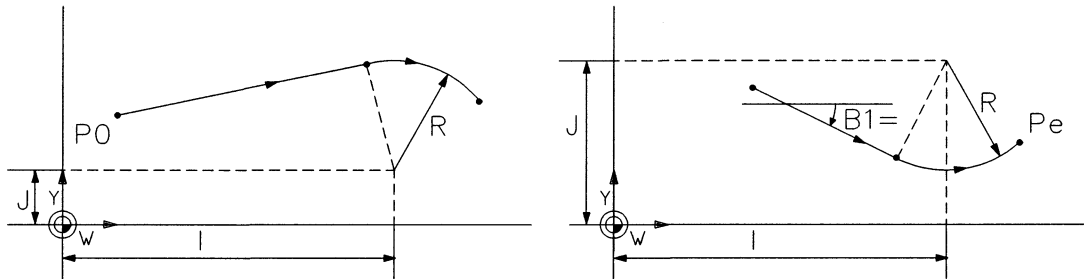
N.. G2/G3 I.. J.. {R..} J1=1/2
 N.. G3/G2 R..
 N.. G2/G3 I.. J.. R..

TWO TANGENT GEOMETRY ELEMENTS

TANGENCY INDICATOR

With the word R1=0 in the first block is indicated, that a line is tangent to a circle or a circle tangent to a line or another circle.

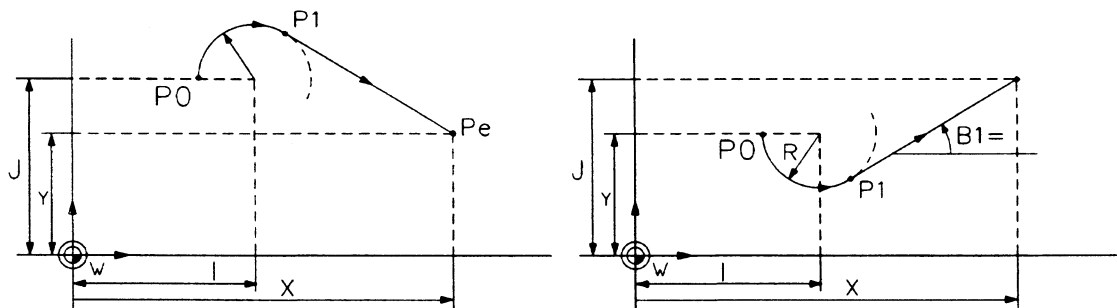
TANGENT LINE AND CIRCLE OR CIRCLE AND LINE



NB9755

Line tangent to circle

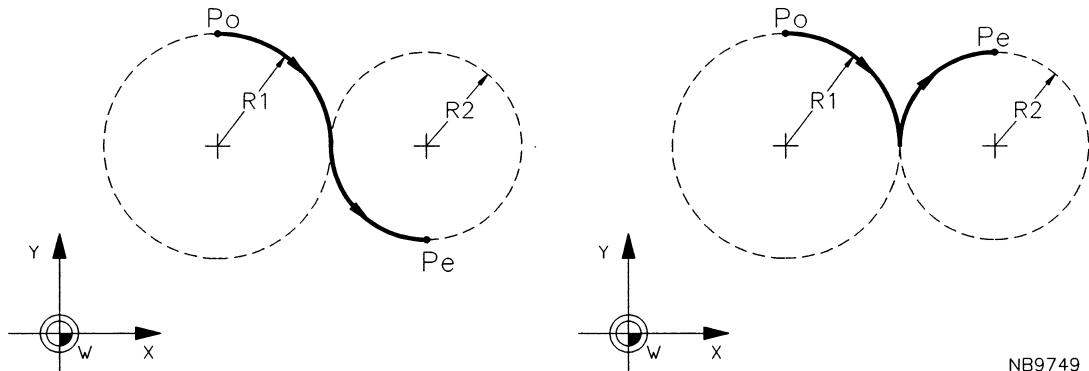
N.. G1 {B1=..} or {support point} R1=0
 N.. G2/G3 I.. J.. R..



NB9756

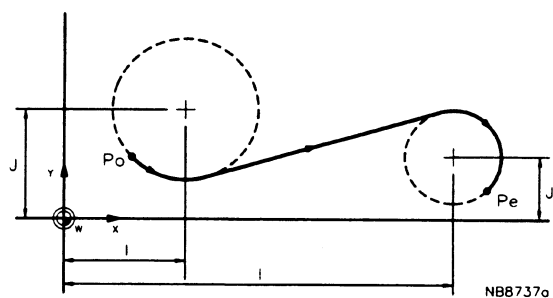
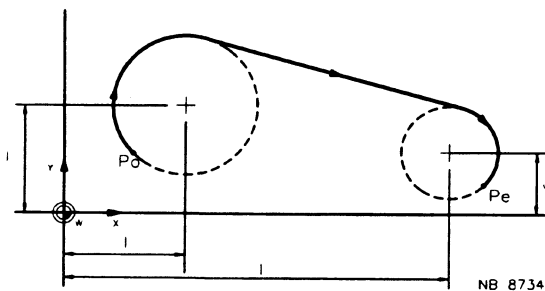
Circle tangent to line

N.. G2/G3 I.. J.. (R..) R1=0
 N.. G1 {B1=..} or {support point} or {end point}



Two tangent circles

N.. G2/G3 I.... {R..} R1=0 J..
 N.. G2/G3 I.... R.. J..



Common tangent line from two circles

N.. G2/G3 I.. J.. {R..} R1=0
 N.. G1 R1=0
 N.. G2/G3 I.. J.. R..

CONNECTING CIRCLES

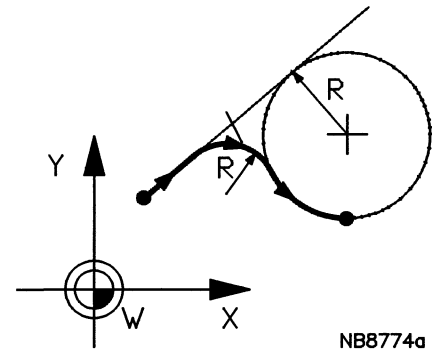
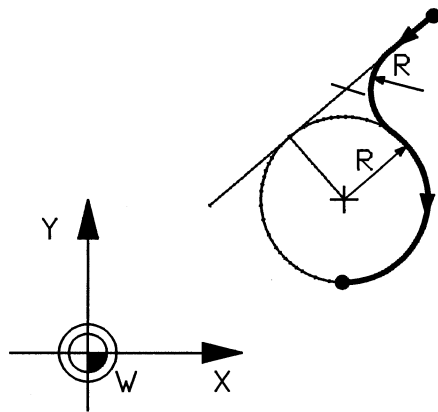
A connecting circle is:

-always tangent to the geometry elements (line or circle) from the previous block and the next one.

-programmed with:

-G2 or G3 indicating the direction of movement,
 -the radius (R-word) of the circle.

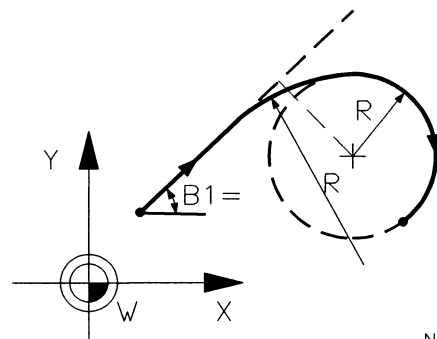
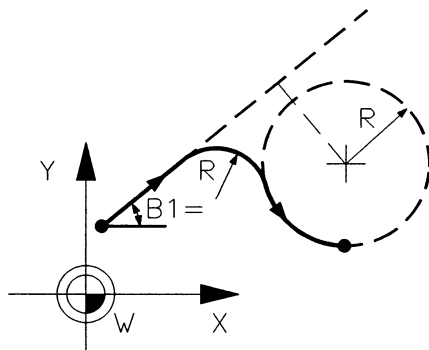
A CONNECTING CIRCLE BETWEEN LINE AND CIRCLE OR CIRCLE AND LINE



NB8774a

Line tangent to circle

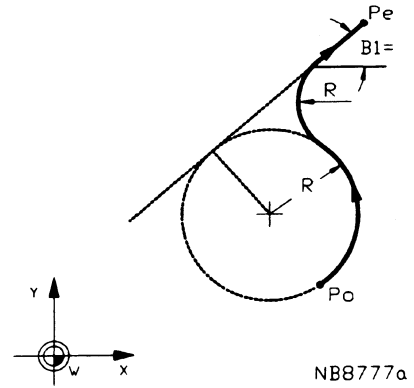
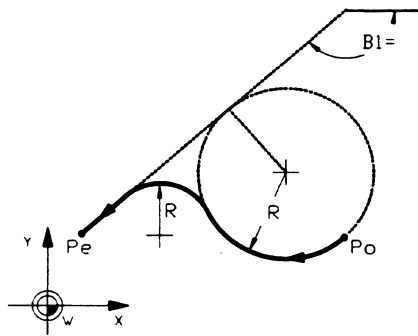
N.. G1 {B1=..} or {support point} R1=0
 N.. G3/G2 R..
 N.. G2/G3 I.. J.. R..



NB9750

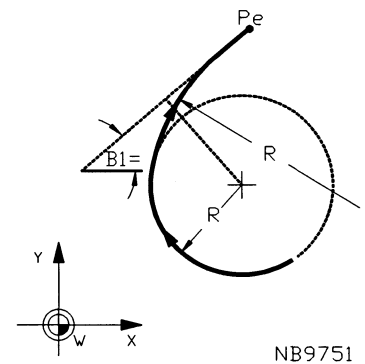
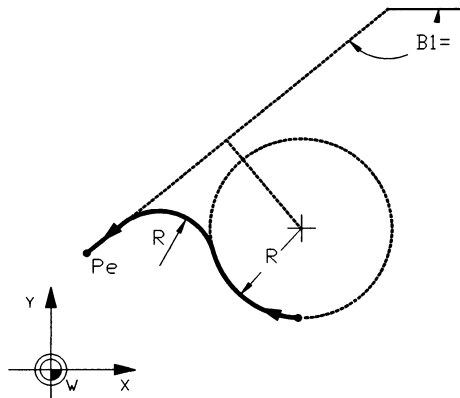
Line does not meet a circle

N.. G1 {B1=..} or {support point}
 N.. G3/G2 R..
 N.. G2/G3 I.. J.. R..



Circle tangent to line

N.. G2/G3 I.. J.. R.. R1=0
 N.. G3/G2 R..
 N.. G1 {B1=..} or {support point} or {end point}

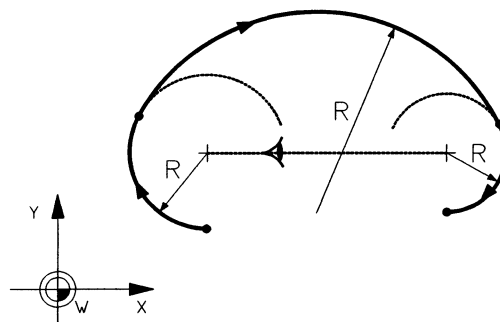
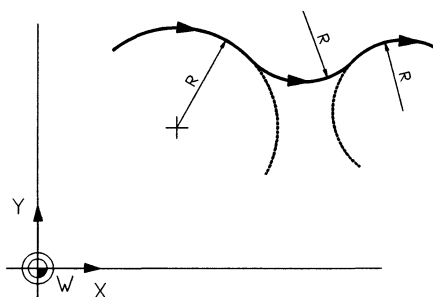


Circle does not meet the line

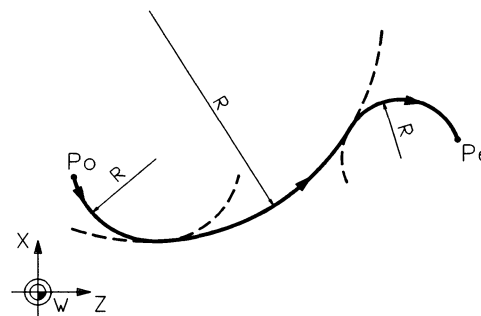
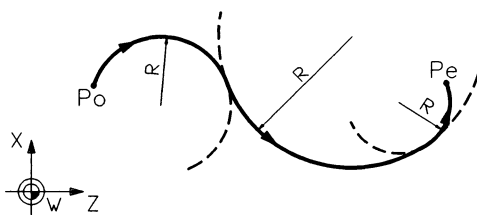
N.. G2/G3 I.. J.. R..
 N.. G3/G2 R..
 N.. G1 {B1=..} or {support point} or {end point}

A CONNECTING CIRCLE BETWEEN TWO CIRCLES OUTSIDE EACH OTHER

To insert a connecting circle between two circles outside each other which do not meet. The direction of rotation on the three circles indicates the type of connecting circle.



NB9752



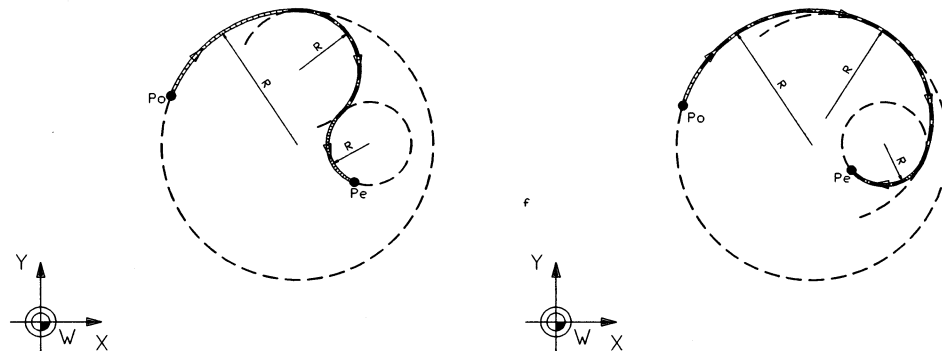
NB9753

For all cases the same format is available:

N..	G2/G3	I..	J..	{R..}
N..	G3/G2	R..		
N..	G2/G3	I..	J..	R..

A CONNECTING CIRCLE BETWEEN TWO CIRCLES OF WHICH ONE CIRCLE INSIDE THE OTHER ONE

To insert a connecting circle between a circle inside the other one which do not meet. The direction of rotation on the three circles indicates the type of connecting circle.



NB9754

For both cases the same format is available:

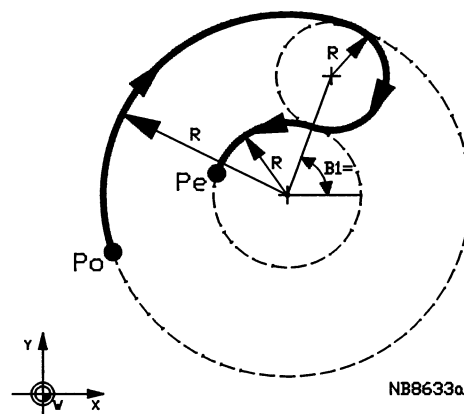
N..	G2/G3	I..	J..	{R..}
N..	G2/G3	R..		
N..	G2/G3	I....	R..	

A CONNECTING CIRCLE BETWEEN TWO CONCENTRIC CIRCLES

Two concentric circles are a very special case of one circle inside the other one. In this case the centre points of both circles coincide. The word B1=.. which indicate the angle with the main axis of the line through the centre point of the concentric circles and the connecting circle, is used as additional information and has to be inserted in the block with the connecting circle.

The formats are:

Radius of the connecting circle is known



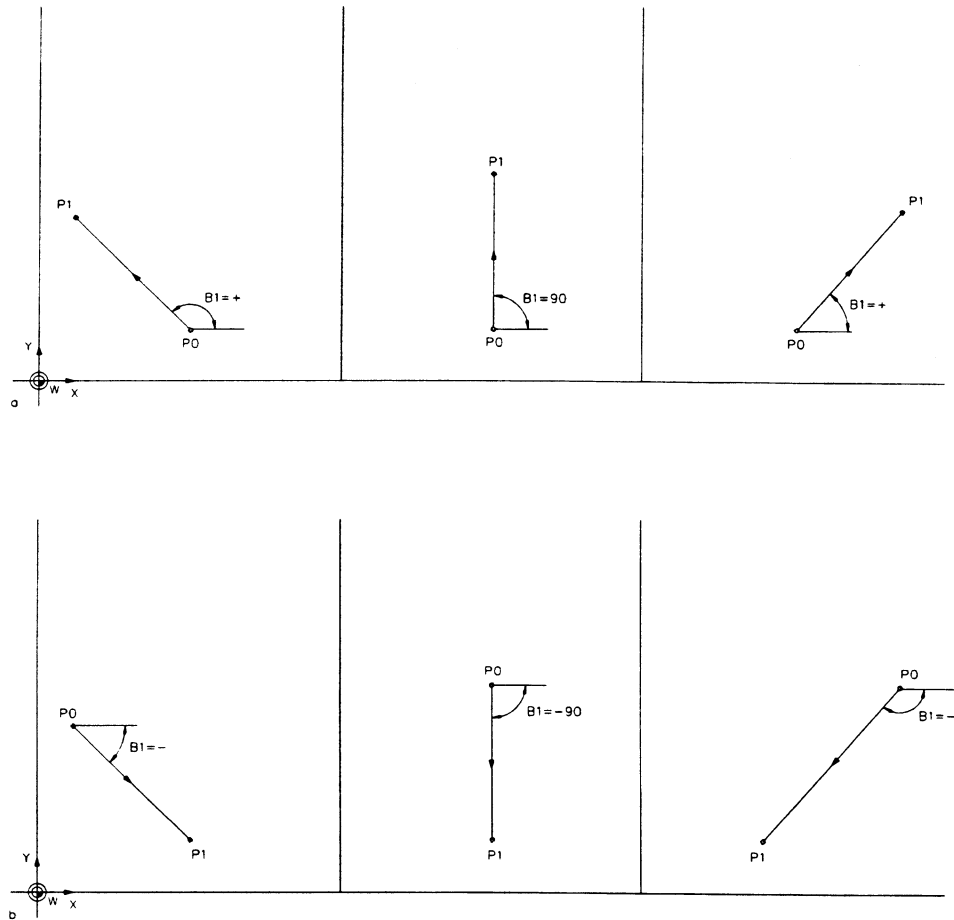
NB8633a

Two concentric circles

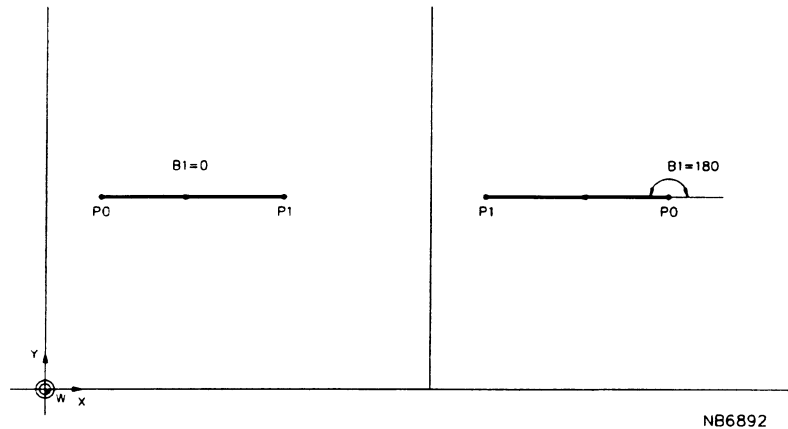
N..	G2/G3	I..	J..	{R..}
N..	G2/G3	R..	B1=..	
N..	G2/G3	I..	J..	

PROGRAMMING THE ANGLE B1=

In a lot of cases a linear movement has to be programmed with the angle which the line makes with the main axis. The angle is programmed with the word B1=.. The angle should be programmed in the direction of movement which means that one should look from the start point of the first movement to the end point of it. The sign of the angle contains the direction of movement and can be seen from the illustration.



NB6891



Note: It is important that the angle is programmed correctly, otherwise the wrong intersection point can be chosen.

END POINT An end point is programmed with:

- the absolute cartesian coordinates X and Y
- the polar coordinates B2= and L2=
- a previously defined point P or P1=

In some cases the intersection point of two elements is known from the drawing and can be programmed as an end point. It is still possible to insert a chamfer between linear movements or a rounding between intersecting elements.

If the intersection point of two lines is programmed as an end point, this point is assumed to be the start point of the next movement and can be programmed with one (X.. or Y..) two coordinates (X.. and Y..) one coordinate and angle (X.. or Y.. and B1=..).

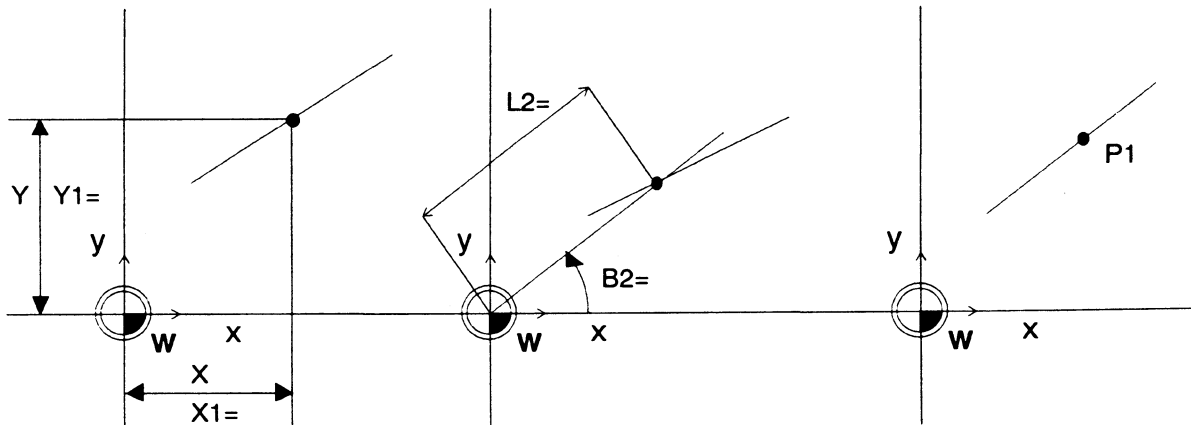
A horizontal or vertical line can also be programmed with one coordinate only. The other coordinate is picked up from the previous blocks. If the start point of the first element is not known, the angle which the line makes with the main axis, has to be added to the block.

CIRCLE CENTRE POINT

The circle centre is programmed with either the absolute cartesian coordinates (I, J, K..) or its polar coordinates (B3=, L3=..).

SUPPORT POINT

When end point coordinates are unknown, another point on the same line can be used to support the calculations of the end point.



NB8010

X.. Y.. I1=0 B2= L2= I1=0 P1 I1=0 X1=.. Y1=

Defining a support point.

Four formats are available:

N.. G1 {B1=..} X1=.. Y1=..

N.. G1 {B1=..} X.. Y.. I1=0

N.. G1 {B1=..} B2=.. L2=.. I1=0

N.. G1 {B1=..} P.. I1=0

Any point on the line can be used as a support point. Its coordinates can be programmed with:

- the absolute cartesian coordinates X1=, Y1= or X, Y
- the polar coordinates B2=, L2=
- a previously defined point P or P1=

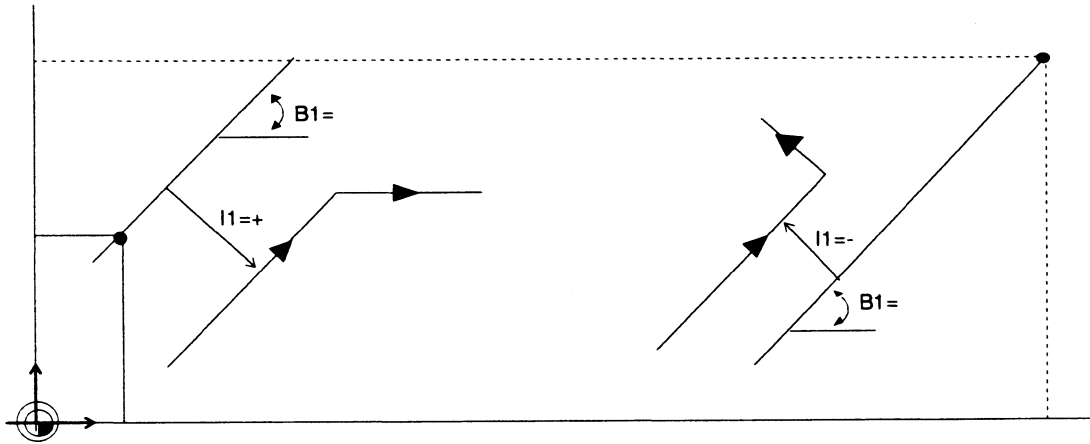
- Note:
1. If the end point coordinates can not be used due to the effect that the tool moves to the end point, a support point with I1=0 has to be programmed.
 2. If a support point is programmed and the line is not yet completely defined, the angle B1=.. which the line makes with the main axis, has to be programmed too. If a block contains too much information, an error message is displayed.

PARALLEL LINE

Sometimes a line is drawn parallel to a known line. The distance between the required line and the known one is programmed with the word I1=. The word I1= has a sign:

I1=+...: the line to the **right** of the existing line

I1=-...: the line to the **left** of the existing line



NB9022

Defining a parallel line

The following formats are available:

N.. G1 B1=.. X.. Y.. I1=+/-..

N.. G1 B1=.. B2=.. L2=.. I1=+/-..

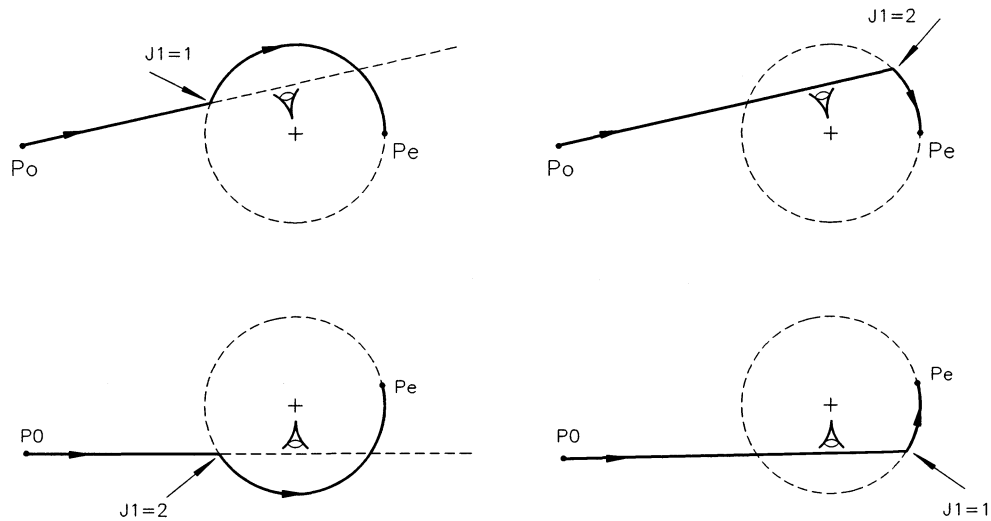
N.. G1 B1=.. P.. P1=.. I1=+/-..

Any point on the existing line can be used. Its coordinates can be programmed with:

- the absolute cartesian coordinates X, Y
- the polar coordinates B2=, L2=
- a previously defined point P or P1=

INTERSECTION POINT INDICATOR

When a line or circle or two circles cross each other, there will be two possible points of intersection. A special word (J1=1 or 2) is used to indicate which intersection point's coordinates must be calculated. Two main methods have to be used for determining which intersection point belongs to J1=1 and which one to J1=2.

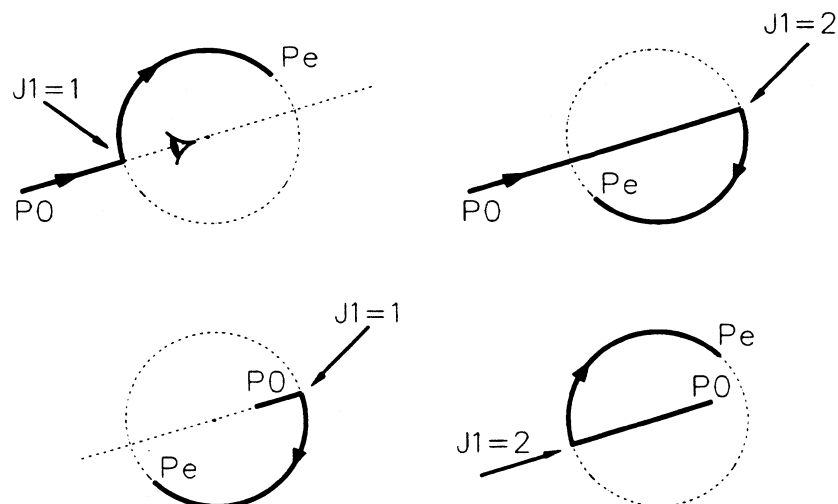
1. when the line goes past the circle's centre

NB9026

From the centre of the circle look at the line. The J1=1 intersection will be on the left and J1=2 intersection on the right of the perpendicular.

2. when the line goes through the centre of the circle**2.1 line intersects circle**

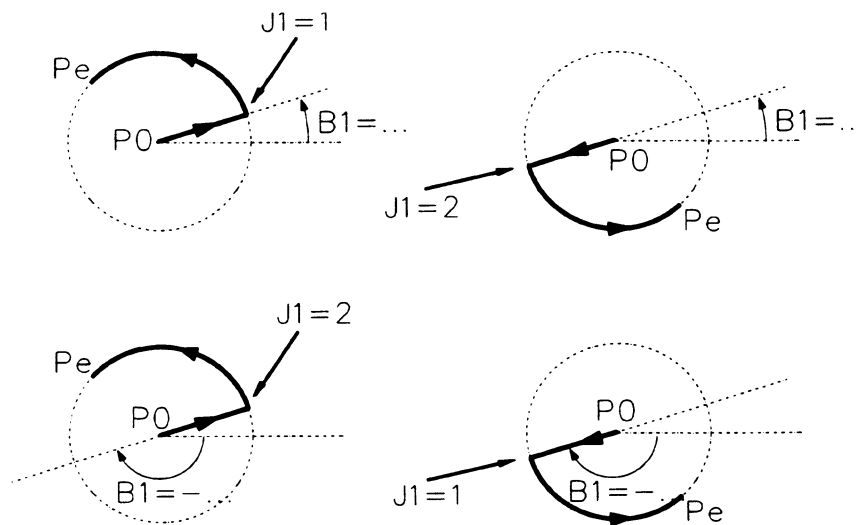
a. start point of the line is not in the circle centre



NB9028

The intersection point closest to the start point is J1=1; the other point is J1=2

b. start point of line is in the circle centre

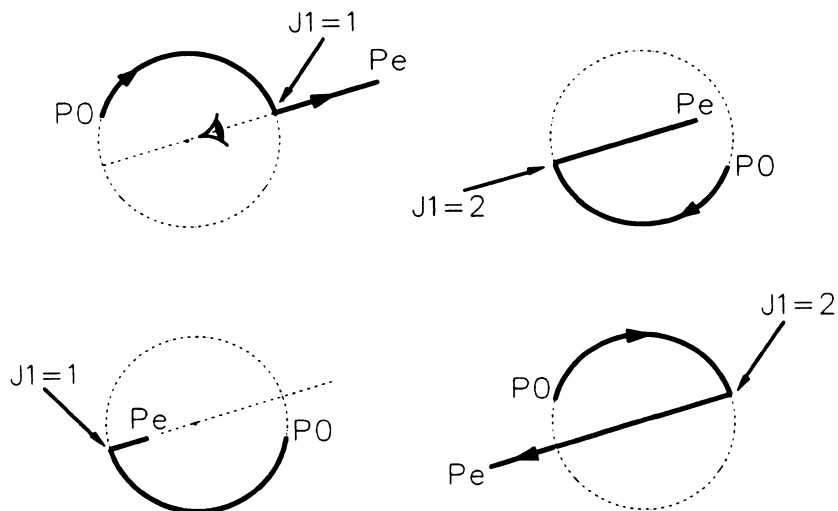


NB9030

The intersection point in the direction of movement on the line defines J1=1; the other point is J1=2.

2.2 circle intersects line

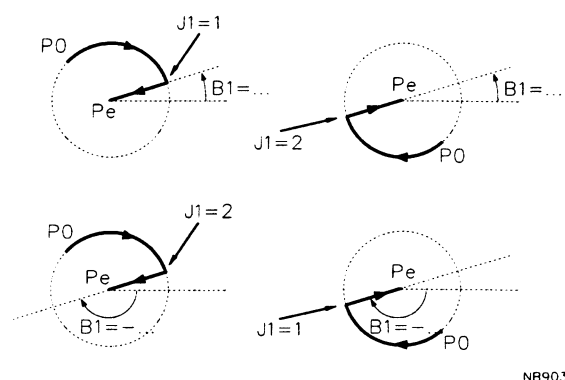
a. end point of the line is not in the circle centre



NB9029

The intersection point closest to the end point is J1=1; the other point is J1=2

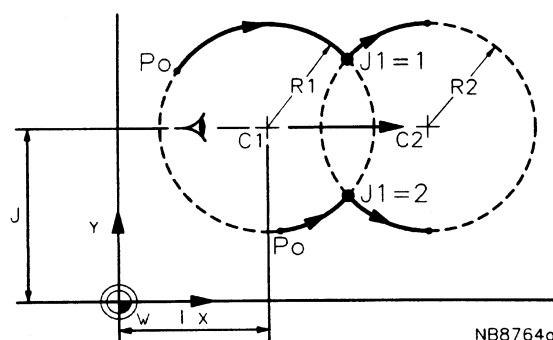
b. end point of line is in the circle centre



NB9031

The intersection point in the direction of movement on the circle defines $J1=1$; the other point is $J1=2$.

3. two circles intersect each other



NB8764a

When two circles intersect each other, the left ($J1=1$) or right ($J1=2$) intersection point is determined by looking from the centre point of the first circle to the centre point of the second one and seeing which intersection point is on the left or on the right from the line through the centres.

Note: Refer to PROGRAMMING THE ANGLE $B1=$ for the meaning of in the direction of movement on the line or circle.

CONTINUOUS AND NON-CONTINUOUS MOVEMENT

With a **continuous** movement the tool moves always in the forward direction. If more than one connecting circle is possible, it depends on the direction of movement on both elements, which connecting circle is automatically taken by the control as default circle. If a short or long arc with a connecting circle is possible, the shorter arc is chosen.

With a **non-continuous** movement:

- the tool can move backwards
- the toolpath can intersect itself
- the longer arc with a circular movement can be taken.

In some milling applications the non-continuous movements have to be used e.g. if the longer arc of a connecting circle should be programmed. In applications like laser cutting the non-continuous movements can be very useful.

Refer to the appendix for programming non-continuous movements.

CANCELLATION

The geometric calculations are cancelled with the function G63. Thereafter complete blocks have to be programmed. In the last block before the cancellation of the geometric calculations an absolute position must be programmed.

DEFAULT MODE

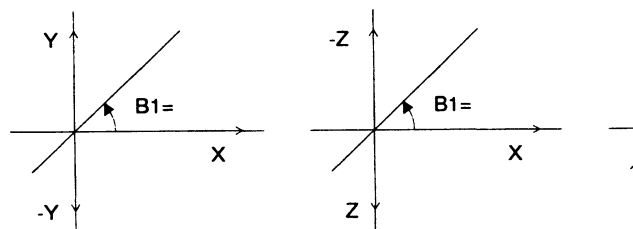
At CLEAR CONTROL the function G63 is automatically activated.

PLANE SELECTION

The geometric calculations are executed in the plane defined by G17 (XY- plane), G18 (XZ-plane) or G19 (YZ-plane).

In the three planes the angle B1= is defined with respect to:

- the + X-axis in the XY- or XZ-plane
- the - Z-axis in the YZ-plane



G17

G18

Angle definition in the different planes

A support point can be programmed with:

- X1= and Z1= in the XZ-plane
- Y1= and Z1= in the YZ-plane

PERMITTED FUNCTIONS

G-functions allowed when the G64-function is active:

G0/G1/G2/G3; G4; G25/G26; G27/G28; G40/G41/G42/G43/G44; G94/G95

Functions **not allowed** when G64 is active

- All G-functions not mentioned in the table above
- Incremental programming (cartesian and polar)
- Helix interpolation
- More than one defined point in a block
- The M-functions M6, M66 and M67

USING MACROS

The use of the geometric calculations is allowed in a macro. All geometry blocks including G63 and G64 must be in the same macro.

USING REPEAT FUNCTIONS

The use of the geometric calculations is allowed in a section of a part program repeated by a G14 or G29. All geometry blocks including G63 and G64 must be in the same section to be repeated.

SCALING, MIRROR IMAGE AND AXES ROTATION

First activating scaling, mirror image or axes rotation and then using the geometric calculations is allowed and results in the required operation on the program section.

Examples

Refer to the chapter Geometric calculations with continuous movements G64.

38. Select negative/positive tool direction G66/G67

Purpose

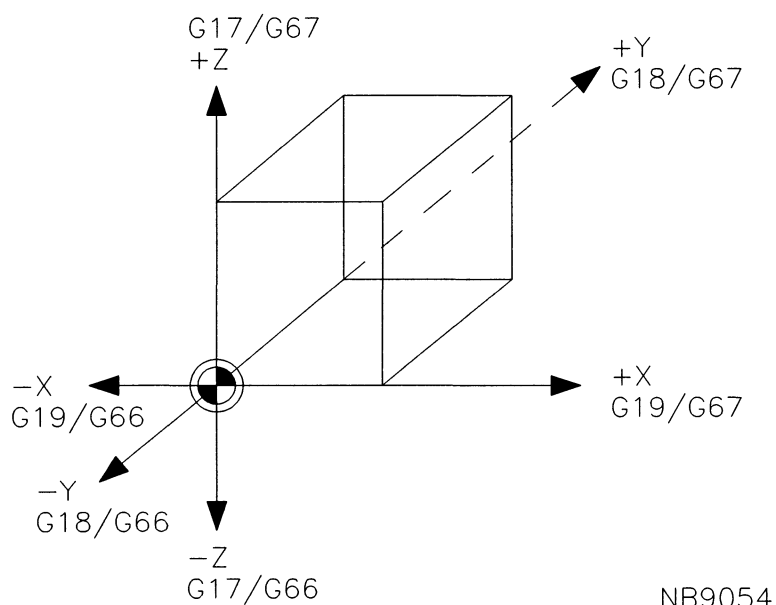
To select the direction in which the tool is pointing:

Tool length compensation in - direction/+ direction G66/G67.

G66: Tool is pointing in the negative direction of the tool axis

G67: Tool is pointing in the positive direction of the tool axis

The use of these functions allows an user always to enter a positive tool length value into the tool memory and the programmer to always look from the tool at the plane for circular interpolation and radius compensation.



Format

Tool pointing in the negative direction of the tool axis:

N... G66

Tool pointing in the positive direction of the tool axis:

N... G67

Associated Functions

G17/G18/G19

Type of function

Modal

Notes and usage

DEFAULT MODE

The function G66 is automatically activated when the CNC is switched on, thus for a tool pointing in the negative direction along the tool axis.

AVAILABILITY

This G66/G67 function is not active in all versions, because it cannot be used on all machine types. It may also be that this function is only possible in G19.

TOOL LENGTH IN THE TOOL MEMORY

The tool length stored in the tool memory, is always a positive value, unless corrections on the length of a standard tool are processed.

TOOL LENGTH COMPENSATION

With G66 active (default mode) the tool length compensation is performed in the negative direction of the tool axis.

With G67 active the tool length compensation is performed in the positive direction of the tool axis.

CIRCULAR INTERPOLATION

To determine the direction of rotation on a circular arc the part programmer looks in the negative (G66) or positive (G67) direction of the tool axis at the plane in which the circle is made. In both cases G2 is used for a clockwise movement and G3 for a counter clockwise movement.

With G67 active the CNC makes the necessary conversions automatically during program execution.

RADIUS COMPENSATION

To determine if the tool is moving on the left or on the right of the workpiece the part programmer looks in the negative (G66) or positive (G67) direction of the tool axis at the plane in which radius compensation is made. In both cases G41 is used for the tool moving on the left and G42 for the tool moving on the right of the work piece.

With G67 active the CNC makes the necessary conversions automatically during program execution.

FIXED CYCLES

If the tool is pointing in the positive direction of the tool axis, the depth of the fixed cycle must be programmed with a positive sign (+) to indicate that the cycle is to be executed in the positive direction of the tool axis. The sign is not automatically inverted.

With the milling cycles (G87 to G89):

- the direction of rotation on the circular arcs is automatically changed in the opposite direction
- the milling direction programmed with the J-word, is not automatically changed.

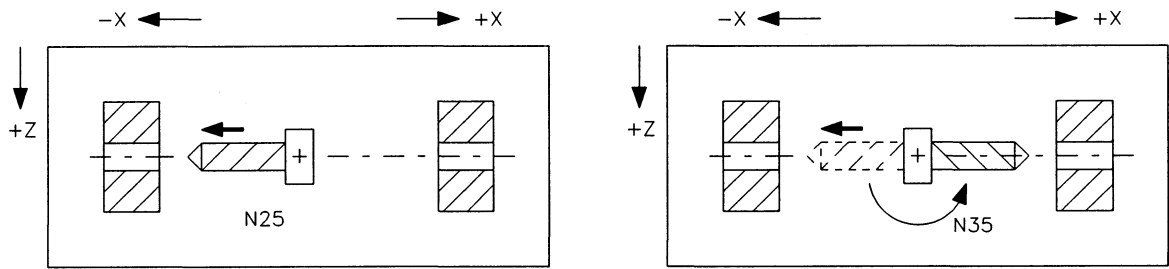
CANCELLATION

Both functions cancel each other. They are not cancelled by CLEAR CONTROL, by M30 or by softkey CANCEL PROGRAM.

RESTRICTION

Do not use the geometry (G64 active) in combination with G67. No error message is displayed, if the geometry is activated after G64. Contour errors may occur, because not all geometry functions are converted properly.

Example



NB9055a

G66 active

G67 active

N25 G1 [End point coordinates]

N30 G67

N35 G1 [End point coordinates]

Explanation

N25: First hole is drilled.

N30: Select tool to point in the positive direction of the tool axis.

N35: Second hole is drilled.

Select negative/positive tool direction G66/G67

39. Inch/Metric programming G70/G71

Purpose

Allows the loading of partprograms which use a different dimensional unit system from that currently active in the CNC.

G70: dimensional units of the part program are in inches.

G71: dimensional units of the part program are in millimetres.

Format

Inch programming:

N... (PROG. NAME) G70

Metric programming:

N... (PROG. NAME) G71

Type of function

Modal.

Notes and usage

DIMENSIONAL UNITS

Units for linear dimensions : .001 mm .0001 inch

Units for feed rate (G94) : .001 mm/min .0001 inch/min

(G95) : .001 mm/rev .0001 inch/rev

Units for cutting speed : 1 m/min 1 feet/min

Note: The cutting speed is used in the technology tables.

ACTIVE SYSTEM OF DIMENSIONAL UNITS

With a machine constant(MC707) is determined which type of dimensional units is used automatically by the CNC at initializing. This is the active system of dimensional units.

Note: The functions G70 and G71 are used at the input level of programmed data. They do not influence the measuring system on the machine tool.

CHANGING THE SYSTEM OF DIMENSIONAL UNITS

If the active system of dimensional units has to be changed, eg. if a part program in the other unit system should be entered via the keyboard, the machine constant setting has to be changed and the control reinitialized. After initializing all dimensions in the memories are divided by 10 (changing from metric to inch) or multiplied by 10 (from inch to metric). Refer also to CNC MEMORIES.

UNIT CONVERSION AT LOADING A PROGRAM FROM A DATA CARRIER

If the CNC detects a G70 or G71 during the loading of a program from a data carrier, the CNC checks if the units used in the program and the active dimensional unit system are the same. If a difference is detected, the CNC converts the coordinates of the linear axes and the feedrates into the equivalents of the active system eg. 'X1' (1 inch) is converted into 'X25.4' (25.4 mm). Also the function G70 or G71 is changed automatically by the control to the opposite function.

Note: Only one type of dimensional units is permitted in a program.

EXECUTING A PART PROGRAM

If a G70 or G71 is not programmed at the beginning of a part program, the CNC assumes that all dimensions are in accordance with the unit system activated on the control.

If one of the functions G70 or G71 is programmed, the CNC checks if the stored program is in the same units as the active system of the control. If a difference occurs, an error is generated.

CNC MEMORIES

The CNC memories in which the tool dimensions, zero offsets, defined points and technology values are stored, must always be in the units of the active dimensional system. If this system is changed, all the stored values must be reentered to their equivalents in the new unit system. The parameter memory is not influenced by a change to the other unit system.

ENTERING A PROGRAM VIA THE KEYBOARD OF THE CONTROL

Programs which are entered into the memory via the keyboard of the control cannot contain a G70 or G71 which conflicts with the active dimensional system. If this is detected, an error message is generated.

Examples**EXAMPLE 1**

CNC active system of units - metric
Partprogram values are in inches.

N9001 (EX.1) G70

N50 G1 X2 Y1.5 F8

Reading block N50 into the part program memory results in storing the coordinates X50.8 Y38.1 and a feed rate of 203.2 mm/min.

EXAMPLE 2

CNC active system of units - inches.
Partprogram values are in millimetres.

N9002 G71

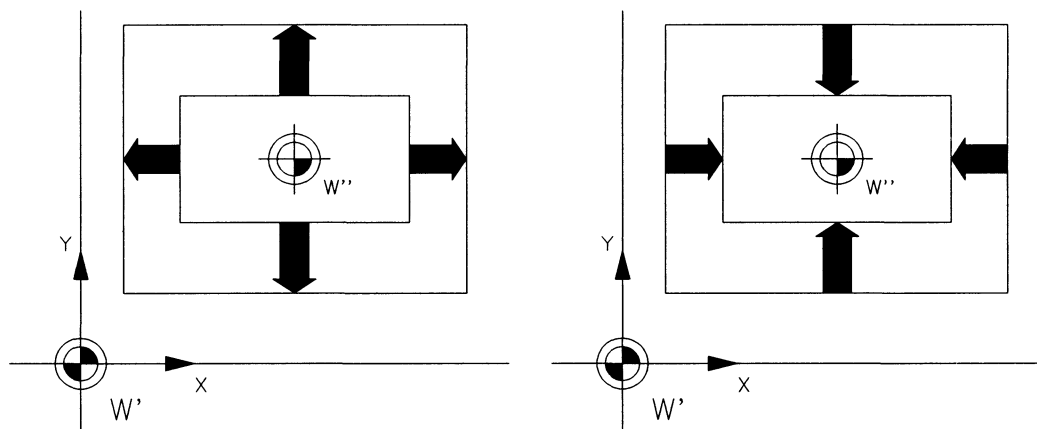
N50 G1 X50.8 Z38.1 F203.2

Reading block N50 into the part program memory results in storing the coordinates X2. Y1.5 and a feed rate of 8 inches/min.

40. Cancel/Activate scaling or mirror imaging G72/G73

Purpose

1. To scale (enlarge or reduce in shape) a group of axis coordinates.

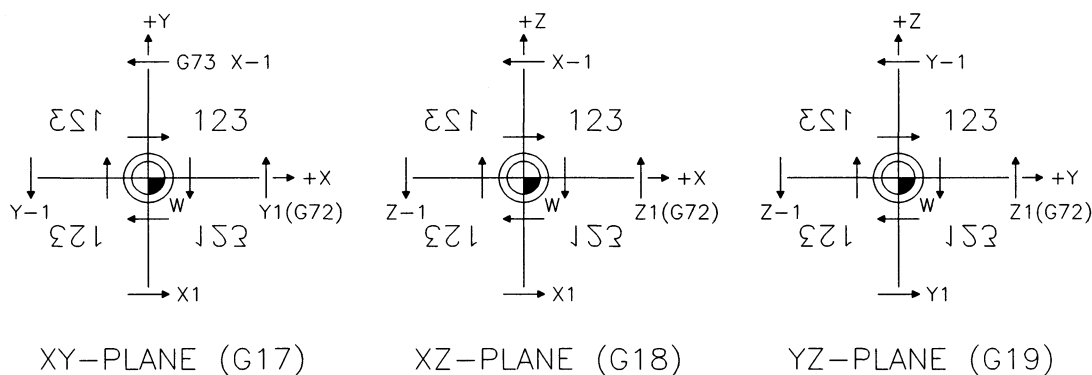


NB8684

Enlargement

Reduction

2. To produce a mirror image of a group of linear main axis coordinates or a change of sign of rotary axis coordinates. (sign inversion)



XY-PLANE (G17)

XZ-PLANE (G18)

YZ-PLANE (G19)

NB8685

G72: To cancel scaling and mirror imaging.

G73: Activate scaling and/or mirror imaging.

Format

To activate scaling

N... G73 A4=...

To cancel scaling

N... G73 A4=1 (factor) or A4=100 (percentage)

To produce a mirror image around an axis or a sign inversion of the axis.

N... G73 {X-1} {Q-1} {2-1} {A-1} {B-1} {C-1}

To cancel mirror image / sign inversion per axis

N... G73 {X1} {Y1} {Z1} {A1} {B1} {C1}

To cancel scaling and mirror image

N... G72

Parameters

G72

No special words

Modal words

H, T1=

G73

Scaling

A4= Scaling factor

Mirroring/Sign change

X X-1:set mirror image / X1:reset

Y Y-1:set mirror image / Y1:reset

Z Z-1:set mirror image / Z1:reset

A A-1:set mirror image / A1:reset

B B-1:set mirror image / B1:reset

C C-1:set mirror image / C1:reset

Modal words

F,S,H, E..=

T1=, T2=,

Associated functions

G92/G93 axis rotation

Type of function

Modal.

Notes and usage

SCALING

SCALING PARAMETER A4=

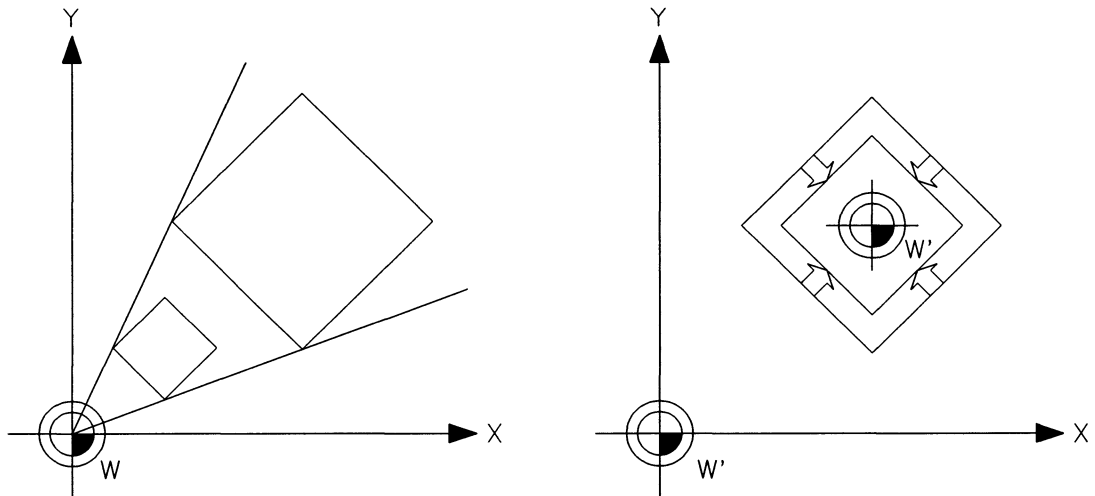
The machine constant MC714 and MC715 determine if the A4= parameter is a factor (format 2.6) or a percentage (format 3.4).

The format of the factor is set with another machine constant.

So a dimension increase of 1.25% is programmed as:

- a factor : G73 A4=1.0125
- a percentage : G73 A4=101.25

GEOMETRIC CENTRE OF THE GROUP OF COORDINATES



NB8688

Scaling about zero point W

Scaling about geometric centre

The scaling function uses the current zero point W as the starting point. If necessary, this point should be set by the use of a G92/G93 zero point shift at the geometric centre of the group of axis coordinates before the scaling operation. This ensures that the coordinates are symmetrically scaled around a fixed point which is not moved out of position by the scaling operation.

PROGRAMMED ZERO POINT SHIFTS (G92/G93)

G92/G93 zero point shifts are scaled if they are present in a group of coordinates to be scaled.

G51-G59 ZERO POINT SHIFTS

The G51-G59 zero point shifts are not influenced by the scaling operation.

SCALING THE TOOL AXIS

With the machine constant for the factor is also determined if scaling is applied to only the axes coordinates in the main plane or to the tool axis as well.

TOOL DIMENSIONS

If the tool axis is to be scaled, the tool length is not scaled.

Tool diameters are not scaled.



NB8689

Before scaling

After scaling - tool is too large

When scaling is to be performed, the programmer must decide if the existing tool diameter is suitable for the different dimensions.

CANCEL SCALING

The scaling is cancelled by:

- G72 mirror image if active, is cancelled too
- Softkey CLEAR CONTROL, M30 and softkey CANCEL PROGRAM, both scaling and mirror image are cancelled
- G73 and the scaling factor: A4=1 or A4=100.

MIRROR IMAGE

SIGN INVERSION

Mirroring around the Y-axis in eg. the XY-plane means changing the sign of the X-coordinate in the opposite sign, thus +X to -X and vice versa (sign inversion). A rotary axis can not be mirrored around an axis, but a sign inversion is still possible, thus +B to -B.

PLANE SELECTION

Mirroring has a meaning in the main plane only

G17 active:

X-1: mirroring around Y-axis

Y-1: mirroring around X-axis

Z-1: sign inversion in tool axis

G18 active:

X-1: mirroring around Z-axis

Z-1: mirroring around X-axis

Y-1: sign inversion in tool axis

G19 active:

Y-1: mirroring around Z-axis

Z-1: mirroring around Y-axis

X-1: sign inversion in tool axis

MIRRORING CIRCULAR MOVEMENTS

When a circular movement is mirrored in one axis, its direction of rotation is also reversed: G2 becomes G3 and G3 becomes G2. This ensures that the tool travels in the correct direction when moving on circular arcs.

TOOL RADIUS COMPENSATION

Tool radius compensation is automatically reversed when mirroring occurs in one axis, for example: G41 becomes G42. This ensures that the tool radius compensation is correctly calculated from the programmed coordinates.

PROGRAMMED ZERO POINT SHIFTS (G92/G93)

G92/G93 zero point shifts are mirrored too, if they are present in a group of coordinates to be mirrored.

G51-G59 ZERO POINT SHIFTS

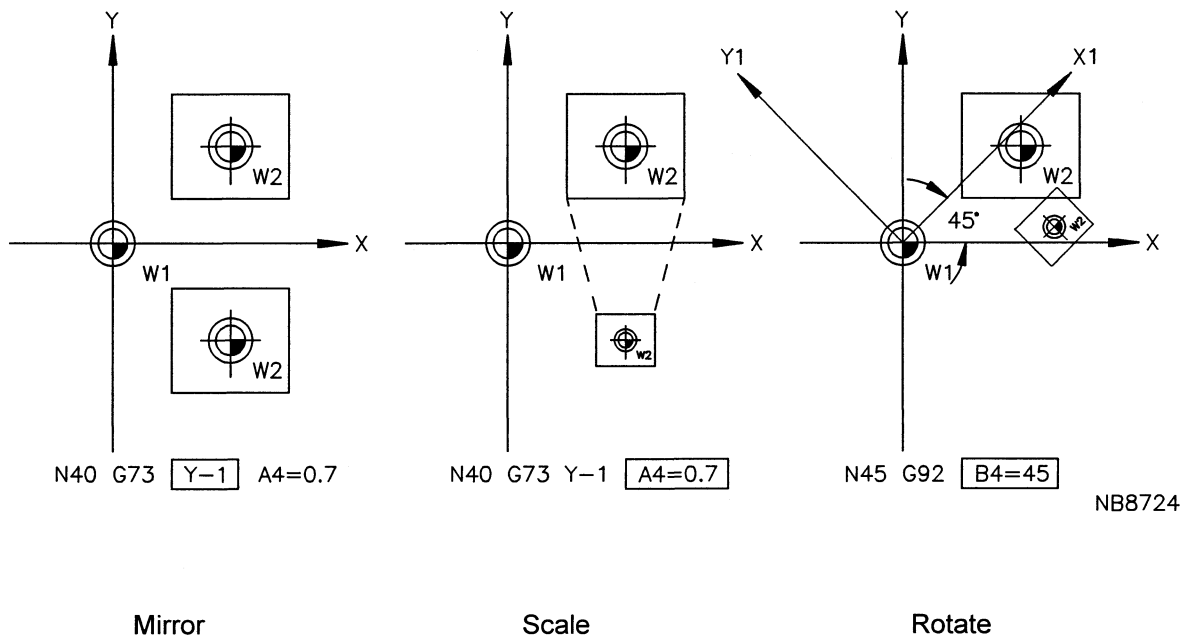
The G51-G59 zero point shifts are not influenced by the mirror operation.

SPINDLE ROTATION

The direction of spindle rotation is not reversed by the mirroring operation. The programmer must therefore consider this fact when deciding which axis coordinates are to be mirrored.

SCALING, MIRROR IMAGE AND ROTATION OF AXES

A group of axes coordinates can be scaled, mirrored and rotated by using a combination of the G73 and G92/G93 functions with the word B4=.



CANCEL MIRROR IMAGE

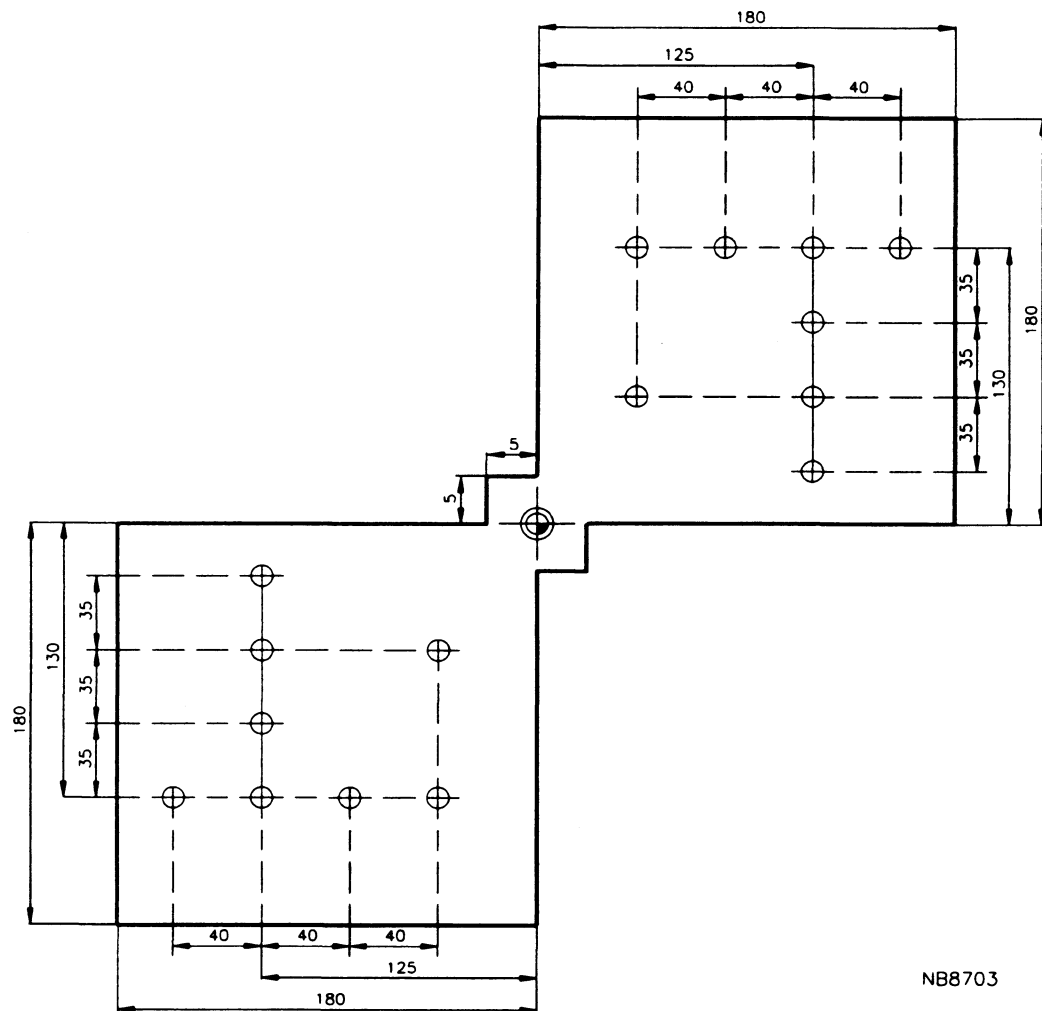
Mirror image is cancelled by:

- G72 a scaling operation, if active, is cancelled too
- Softkey CLEAR CONTROL, M30 and softkey CANCEL PROGRAM, both scaling and mirror image are cancelled
- G73 and the positive sign of the mirrored axis;

eg. X-1 is cancelled by X+1.

Example

Mirror operation



NB8703

N7273 (MIRROR IMAGE OF A POCKET)

N1 G17

N2 G54

N3 S300 T1 M6 (Mill radius 4 mm)

N4 G0 X-5 Y10 Z10 F700 M3

N5 G1 Z-15

N6 G43 Y5

N7 G41

NS G1 X0

N9 G1 Y180

N10 G1 X180

N11 G1 Y0

N12 G1 X5

N13 G1 Y-10

N14 G40

N15 G1 Z10

N16 G73 X-1 Y-1

N17 G14 N1=4 N2=15

N18 G72

N19 S100 T2 M6 (Drill radius 4 mm)

N20 G81 Y10 Z-20 F200 M3

N21 G79 Y60
N22 G79 Y95
N23 G79 Y130
N24 G79 X165
N25 G79 X85
N26 G79 X45
N27 G79 Y60
N28 G73 X-1 Y-1
N29 G14 N1=21 N2=28
N30 G72
N31 G0 Z200 M30

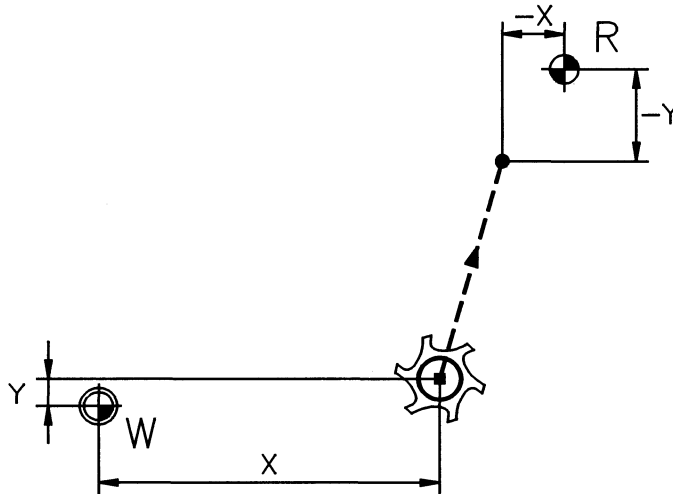
Explanation:

N1: Select XY-plane.
N2: Set program zero point.
N3: Load tool number 1.
N4: Move tool rapidly to programmed position. Set feedrate to 700 mm/min and spindle rotation in clockwise direction.
N5: Feed tool to depth at set feedrate.
N6: Move tool T0 programmed position.
N7: Select tool radius compensation LEFT.
N8-N13: Machine the workpiece.
N14: Cancel tool radius compensation.
N15: Retract tool from workpiece.
N16: Mirror coordinates around X- and Y-axis.
N17: Repeat instructions from block 4 to 15.
N18: Cancel the mirroring operation.
N19: Load tool number 2.
N20: Define drilling cycle and start spindle again.
N21-N27: Execute drilling cycle at programmed points.
N28: Mirror coordinates around X- and Y-axis.
N29: Repeat instructions from block 21 to 27.
N30: Cancel the mirroring operation.
N31: Retract tool in tool axis and end of program.

41. Programmable absolute position G74

Purpose

To execute a rapid traverse movement to a position programmed with coordinates measured from the machine reference point R or machine positions.



Format

N... G74 X.. Y.. Z.. {X1=..} {Y1=..} {Z1=..} {K...} {L...} {K2=...}

Parameters

X,Y,Z Endpoint coordinate
 X1=,Y1=,Z1= Absolute position MC (1-10)
 A,B,C Endpoint angle
 K 0=inpod / 1=inpos / 2=inpoc
 K2= Inpoc window (0: MC, 1-32767 μm)
 L L0:with toollength / L1:without

Modal words

F, S, E...=

Associated functions

G0

Type of function

Non-modal.

Notes and usage

APPLICATION

The G74-function's main application is in programming cycles for tool changers, pallet stations etc., when it is advisable that the programmed coordinates are independent of those used to define the machining of the workpiece.

Note: The absolute position is entered with the addresses X1=..., Y1=..., K2=.. etc. for installation purposes!

END POINT COORDINATES

The end point coordinates can be defined in two different ways:

- 1) X100: relative position to the reference point.
- 2) X100 X1=2: relative position to Home position 2 (MC3146).

For the first axis the machine positions 1 to 10 are describes in machine constants MC3145 -- MC3154. For the second axis MC3245 -- MC3254.

When the actual machine constant is zero, no movement will be done.

STOP BETWEEN BLOCKS (K-word)

All programmed axes move simultaneously during the execution of the G74. The next movement starts once all axes have reached their position. There is a stop between the G74-block and the next one as is usual with rapid traverse movements. (K0 is the default setting). With K1 the stop between the blocks can be avoided.

K0: Allowance is made for a (precise) stop between the movement of the G74 block and the movement in the next block, as is usual for rapid traverse movements.
(K0 is the start setting).

K1: No allowance is made for a stop between the movement of the G74 block and the movement in the next block (corner rounding). The next movement is started when the desired position is nearly reached in all axes.

K2: No allowance is made for a stop between the movement of the G74 block and the movement in the next block. The next movement is started when the desired position is nearly reached in all axes. This position is defined by machine constant (MC136) (K2=0) or by the window size (K2=...).

K2= Display dimensions in mm (0-32.766 mm)

INCREMENTAL MOVEMENTS

If an incremental movement is programmed after a G74 movement, the coordinates are measured from the position stated in the G74-block.

TOOL LENGTH COMPENSATION

In general tool length compensation is not used with the G74 positioning (L0 is default setting). If length compensation is required, 'L1' must be programmed.

RADIUS COMPENSATION

Radius compensation (G41 - G44) must be cancelled before the G74- function is activated.

GEOMETRY FUNCTION

The G64 geometry function must not be active when G74 is used.

ZERO POINT SHIFTS AND ZERO OFFSETS

The active zero point shift and zero offset are temporarily overridden.

AXIS ROTATION AND SCALING

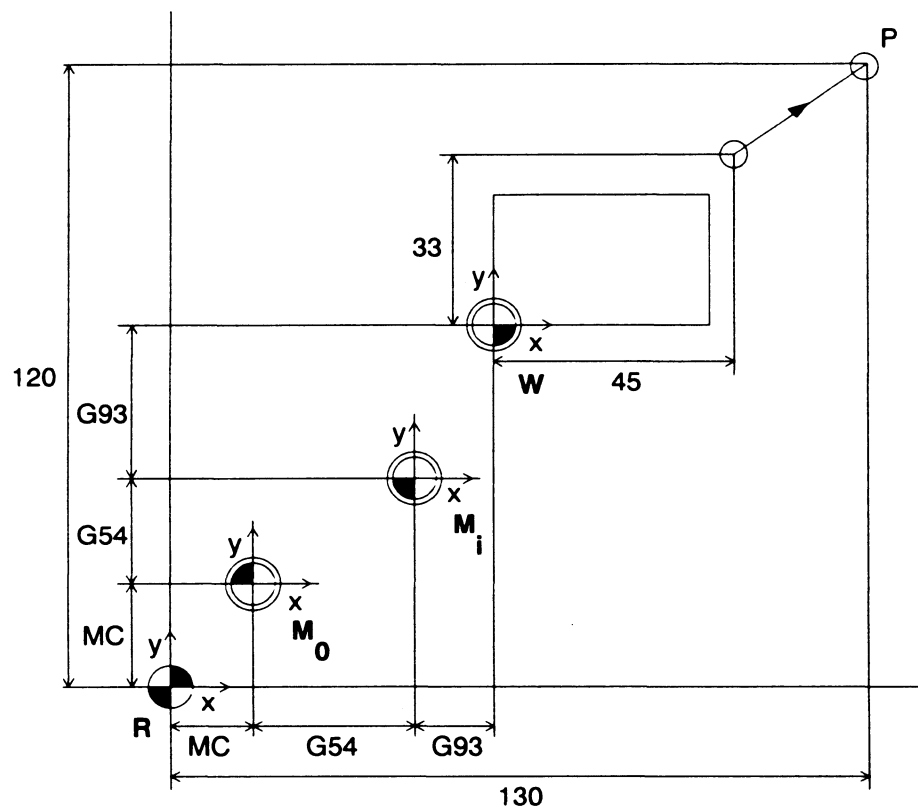
The programmed G74-position is not affected by axis rotation or scaling.

AFTER EXECUTION A G74 BLOCK

All zero points and the tool length compensation (if suppressed) become active again after a G74-block.

The last movement before G74 is activated, must use either the G0 or G1 function. This function is automatically used in the first movement following the G74-block.

Example



NB7980

From point P the coordinates with regard to R are known. The positioning to P is programmed as:

N10 G0 X45 Y33

N11 G74 X130 Y120

N20 G74 X100 X1=1 Y123.456 Z1=10 K2 K2=25.2

X100 X1=1 Relative position to Home position der machine constants (MC3145).

Y123.456 Relative position to the reference point.

Z1=10 (Z0) Position relative to Home position der machine constants (MC3554).

K2 No allowance is made for a stop between the movement of the G74 block and the movement in the next block. The next movement is started when the desired position is nearly reached in all axes. This position is defined by the window size (K2=...).

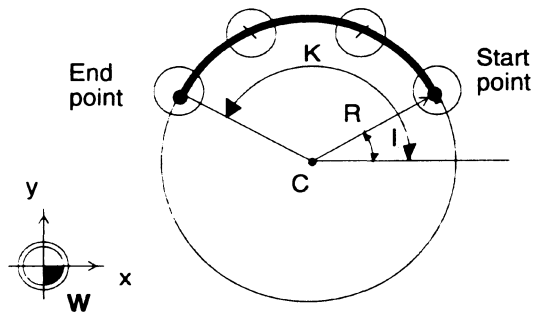
K2= Display dimensions in mm

42. Bolt hole circle G77

Purpose

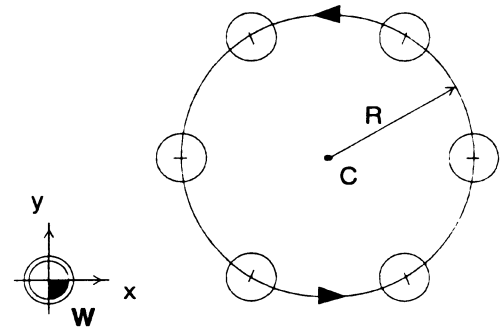
To execute any fixed cycle (G81,G83-G89) at points which are equally spaced on a circular arc or a complete circle.

Format



NB8546

CIRCULAR ARC



NB8547

COMPLETE CIRCLE

Points on an arc

N... G77 [centre point coordinates] R... J... I... K...

Points on a complete circle

N... G77 [centre point coordinates] R... J... I...

Parameters

X,Y,Z Center point coordinate
 A,B,C Endpoint angle
 I Angle to first point
 J Number of points
 K Angle to last point
 P Point nr. centre
 R Circular pattern radius
 B1= Angle
 B2= Polar angle
 L1= Path length
 L2= Polar length
 P1= Point definition nr. for centre
 ENNN= Parameter definition

For absolute and incremental programming

X90=,Y90=,Z90= Absolute centre point
 A90=,B90=,C90= Absolute endpoint angle
 X91=,Y91=,Z91= Incremental centre point
 A91=,B91=,C91= Incremental endpoint angle

Associated functions

G79, G81, G83-G89

Wordwise absolute/incremental programming (X90=..., X91=...)

Type of function

Non-modal.

Notes and usage

DIRECTION OF EXECUTING THE POINTS

The fixed cycles are always executed in a counter clockwise direction along the circular path.

I AND K ANGLES

Maximum angle: +/- 360 degrees.

The angle is programmed in degrees and decimal parts of a degree in steps of .001 degrees.

PLANE FOR THE PATTERN

The circular pattern is situated in the plane defined by the active G-function for plane selection (G17, G18 or G19)

The defined cycle is executed in the tool axis which is perpendicular to the active main plane.

EXECUTING THE FIXED CYCLE

The cycles are executed on the points of the circular are as if G79- blocks are programmed. Refer to the function G79 for additional information about executing the fixed cycle.

ROTATED POCKET OR GROOVE (B1=)

A previously defined pocket or groove (G87 or G88) can be rotated about an angle. The centre of rotation is the point used in the G77 block to program the location of the pocket or groove.

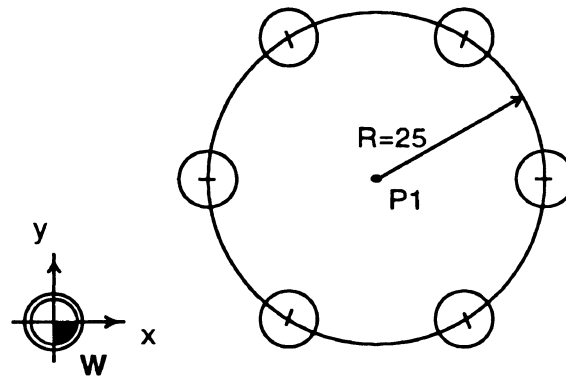
The angle is programmed with the word B1= in degrees and decimal parts thereof and ranges from - 360° to 360°.

The angle is measured with the X-axis (G17 and G18) or the -Z-axis (G19).

Three possibilities are available:

1. B1= not programmed in the G77 block
In this case the sides of the pocket or groove are parallel to the main axes.
2. B1=0
In this case the axis of each pocket or groove is radial, thus lies in the direction of the radius from the centre of the circle to the point on the circle. Refer to example 3 for programming this case.
3. B1<>0
In this case B1= indicates the angle which the pocket or groove makes with the radius to the centre of the pocket. Refer to example 4 for programming this case.

Note: The word B1= has two meanings in a G77 block. Either it is the angle for rotating a pocket or groove or it is used to program the coordinates (B1=, L1= or X/Y with B1=) for the position of the centre of the circle.

Examples**EXAMPLE 1. Fixed cycle on a complete circle**

NB5820

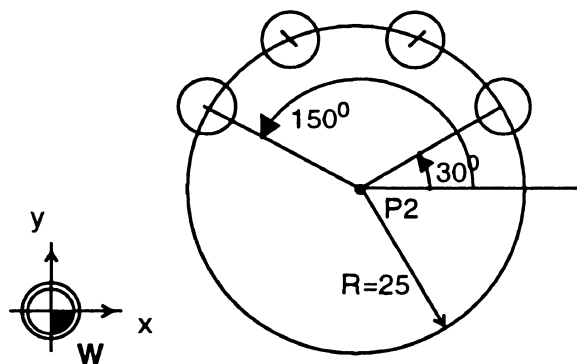
```

N41 G78 P1 X... Y... Z...
:
N49 T1 M6
N50 G81 Y1 Z-10 F100 S1000 M3
:
N60 G77 P1 R25 I0 J6

```

Explanation

N41: Definition of circle centre point (P1)
 N49: Load tool 1 (a drill)
 N50: Fixed cycle definition
 N60: Execute the fixed cycle on six points of the complete circle.

EXAMPLE 2. Fixed cycle on an arc

NB5819A

```

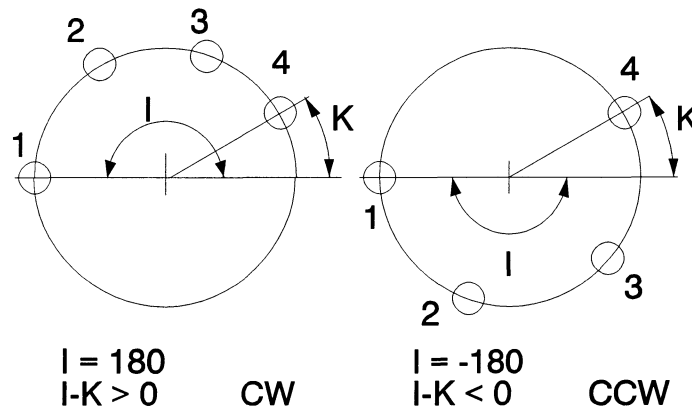
N40 G78 P2 X... Y... Z...
:
N49 T1 M6
N50 G81 Y1 Z-10 F100 S1000 M3
:
N60 G77 P2 R25 I30 K150 J4

```

Explanation

- N40: Definition of circle centre point (P2)
 N49: Load tool 1 (a drill)
 N50: Fixed cycle definition
 N60: Execute the fixed cycle on four equally spaced points on the circular arc, starting from 30 degrees and ending at 150 degrees.

EXAMPLE 3. Direction holes on a arc

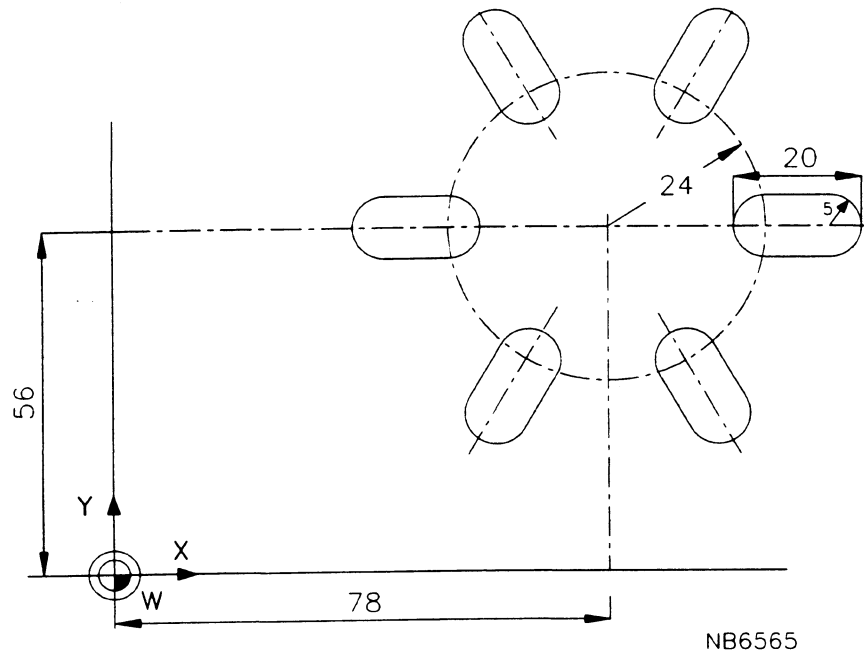


N50 G81 Y1 Z-10 F100 S1000 M3
 N60 G77 X0 Y0 Z0 R25 **I180 K30** J4
 N70 G77 X0 Y0 Z0 R25 **I-180 K30** J4

Explanation:

- N50 : Cycle definition
 N60 : Cycle repeating four times on an arc; Start on 180 degrees, End on 30 Grad in Clockwise (CW) direction.
 N70 : Cycle repeating four times on an arc; Start on -180 degrees, End on 30 Grad in Counterclockwise (CCW) direction.

EXAMPLE 4. Radial grooves



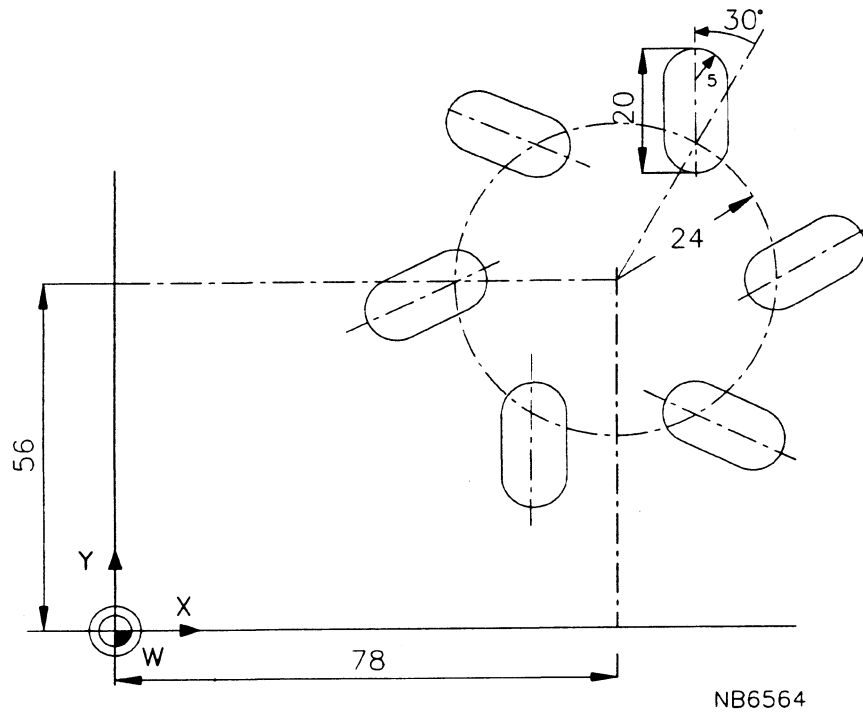
NB6565

N55 G17
 N60 T1 M6
 N65 G88 X20 Y10 Z-10 B1 F100 S1000 M3
 N70 G77 X78 Y56 Z0 R24 IO J6 B1=0

Explanation

- N55: Define the main plane for the grooves
 N60: Load tool 1, a mill with a radius of 4.8 mm
 N65: Define the groove as if its sides are parallel to the X- and Y-axis
 N70: The radial grooves are milled.
 This block contains:
- the centre of the bolt hole circle (X78, Y56, Z0),
 - its radius (R)
 - the angle which the radius of the first point makes with the X-axis (I)
 - the number of holes on the circle (J)
 - B1=0 to indicate that the grooves are radial.

EXAMPLE 5. Rotated groove



N55 G17
 N60 T1 M6
 N65 G88 X20 Y10 Z-10 B1 F100 S1000 M3
 N70 G77 X78 Y56 Z0 R24 IO J6 B1=30

Explanation

N55: Define the main plane for the grooves
 N60: Load tool 1, a mill with a radius of 4.8 mm
 N65: Define the groove as if its sides are parallel to the X- and Y-axis
 N70: The rotated grooves are milled.
 Refer to the previous example for an explanation of the addresses.
 Only the word B1= has a different meaning.

43. Point definition G78

Purpose

Allows the coordinates of a point to be defined just once in a program. When a movement to the point is required, only the point number has to be programmed, not the points coordinates.

Format

N... G78 P... [Coordinates of points position]

Parameters

X,Y,Z Point coordinate
A,B,C Point angle
B2= Polar angle
L2= Polar length
P Point definition number

Modal words
E...=

Associated functions

G0, G1, G2/G3, G45, G46, G77, G79, G92/G93, G145

Type of function

Non-modal

Notes and usage

COORDINATES

Only cartesian coordinates measured from the active program zero point W or polar coordinates (B2=, L2=..) in the main plane can be used.

G-FUNCTIONS WITH PRE-DEFINED POINTS

The following G-functions can only contain one defined point in their program block: G2/G3, G77, G92, G93.

The following G-functions can have a maximum of four defined points in their program block: G0, G1 and G79.

USING A PRE-DEFINED POINT

The format for using a predefined point is as follows:

N... G... P..., where P... represents the number of the point in the point memory.

Other formats are possible:

N... G79 P4=2 P2=10 P3=1 P1=5

N... G79 P1=E5 P2=E1

The P address can also be programmed with an index. The index value indicates the priority in the execution sequence. Index numbers 1 up to 4 are available (1=highest priority, 4=lowest priority). The number of the point in the point memory is entered behind the = sign.

Another option would be parametrized entry of the point definition. In this event the index also indicates the priority.

POINT MEMORY

A maximum of 255 defined points can be stored in the CNC's Point Memory; this maximum is set by a Machine Constant.

Cancellation

A defined point's coordinates remain active until:

- the point is redefined again by another G78-block;
- the point memory is changed or cleared by the user;
- a data carrier with defined points is read in.

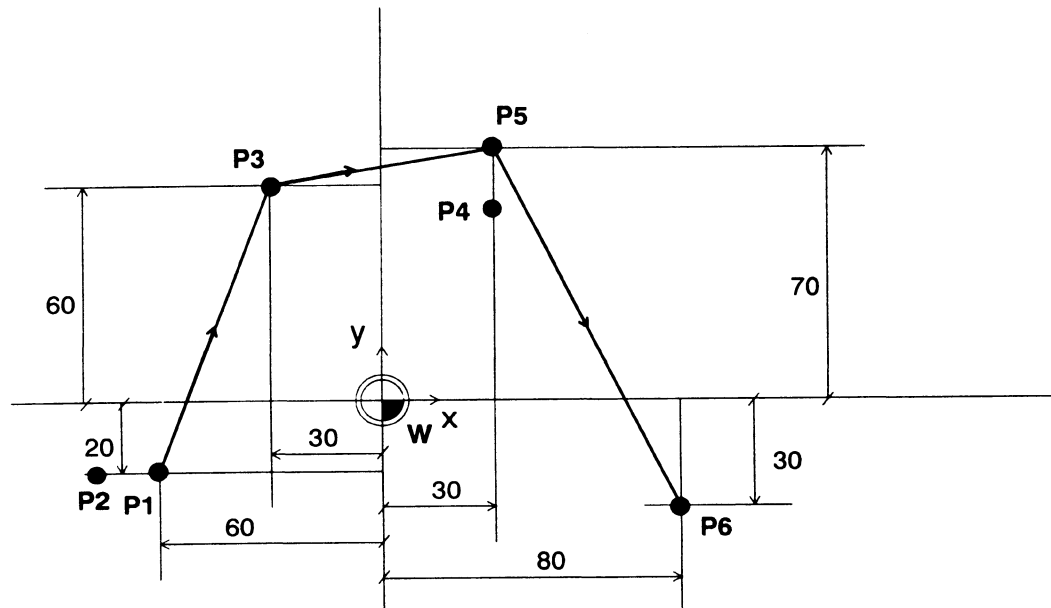
The Point Memory is not affected by CLEAR CONTROL.

RESTRICTIONS

Only one point can be specified in each G78-block, no other words are permitted.

Examples

EXAMPLE 1.



NB8548

```

N10 G78 P1 X-60 Y-20
N11 G78 P2 X-70 Y-20
N12 G78 P3 X-30 Y60
N13 G78 P4 X30 Y50
N14 G78 P5 X30 Y70
N15 G78 P6 X80 Y-30

```

```

N90 G0 P1=1
N91 G1 P1=3 P2=5 P3=6 F1000

```

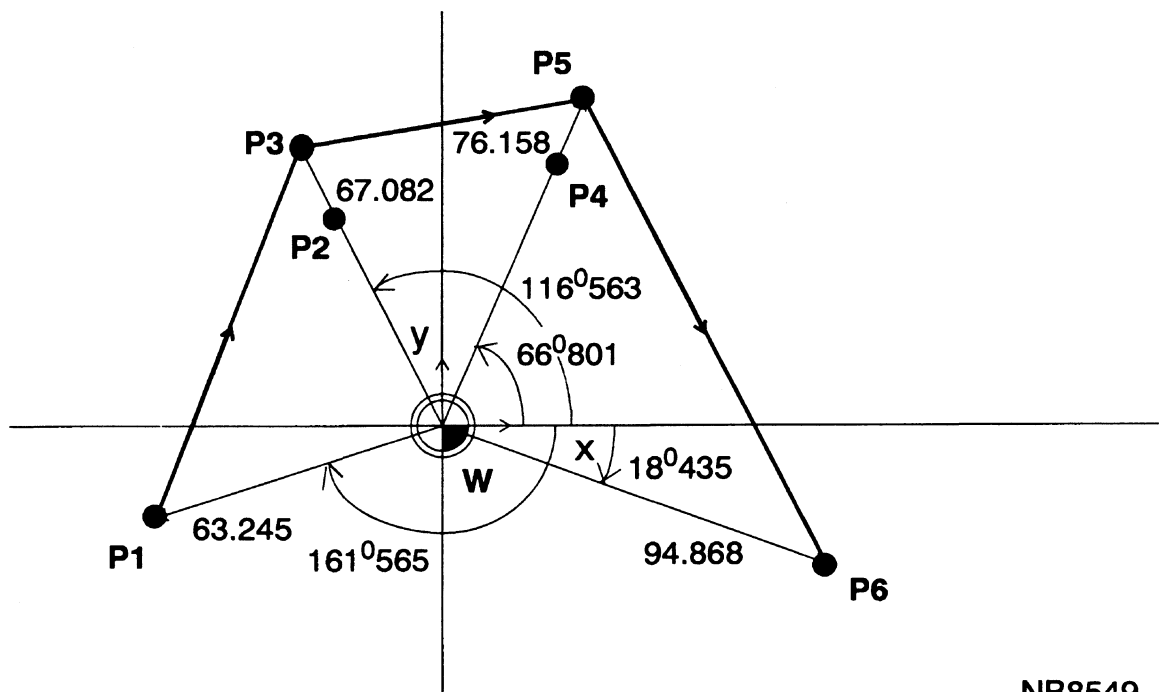
Explanation:

N10-N15: The points are defined.

N90 : Move tool rapidly to position defined by P1.

N91 : Move tool at set feedrate, first to P3, then to P5 and finally to P6.

EXAMPLE 2. Using polar coordinates



NB8549

```

N10 G78 P1 B2= -161.565 L2= 63.245
N11 G78 P2 B2= 116.563 L2=65
N12 G78 P3 B2= 116.563 L2= 67.082
N13 G78 P4 B2= 66.801 L2=72
N14 G78 P5 B2= 66.801 L2= 76.158
N15 G78 P6 B2= -18.435 L2= 94.868

```

```

N90 G0 P1 =1
N91 G1 P1=3 P2=5 P3=6 F1000

```

Explanation:

N10-N15: The points are defined.

N90 : Move tool rapidly to position defined by P1.

N91 : Move tool at set feedrate, first to P3, then to P5 and finally to P6.

44. Activate cycle G79

Purpose

To activate any fixed cycle which controls hole operations (G81, G83 to G86) or milling operations (G87 to G89) at programmed positions.

Format

N... G79 [Axis coordinates] {B1=...}

Parameters

X,Y,Z Point coordinate
 A,B,C Point angle
 B1= Angle
 L1= Path length
 B2= Polar angle
 L2= Polar length
 P Point definition number (P1-P4)
 P1=,P2=,P3=,P4= Point definition number

For absolute and incremental programming
 X90=,Y90=,Z90= Absolute centre point
 A90=,B90=,C90= Absolute endpoint angle
 X91=,Y91=,Z91= Incremental centre point
 A91=,B91=,C91= Incremental endpoint angle

Modal parameters
 F,S,T,M,H, E...=
 Ti=, T2=

Associated functions

G77, G81, G83-G89
 Wordwise absolute/incremental programming (X90=..., X91=..)

Type of function

Non-modal.

Notes and usage

EXECUTING DEFINED FIXED CYCLES

The positions where a previously defined fixed cycle is to be executed, are programmed in the G79-blocks which follow the fixed cycle definition.

A fixed cycle for a hole operation (G81,G83-G86) is executed in the tool axis which is perpendicular to the main plane defined by the G- function for plane selection (G17, G18 or G19).

A fixed milling cycle is executed in the main plane defined by the G- function for plane selection (G17, G18 or G19).

The first G79-block which follows a defined fixed cycle must contain a tool axis coordinate.

The direction of the depth operation is programmed in the fixed cycle block with the sign of the Z-word indicating the total depth.

POSITIONING LOGIC

To reduce the possibility of collision between tool and workpiece the positioning logic is available. Refer to G0 POSITIONING LOGIC for additional information.

In addition, the programmer must ensure that the tool can not collide with any device holding the workpiece in position.

SPINDLE ROTATION

If the spindle is not rotating when a G79-block is activated, an error message is generated and program execution stopped.

RADIUS COMPENSATION

When a G79-block is activated, radius compensation is cancelled by the CNC automatically generating the G40-function.

If radius compensation is required after a G79-block, the appropriate function (G41 -G44) must be programmed.

ROTATED POCKET OR GROOVE (B1=)

A previously defined pocket or groove (G87 or G88) can be rotated about an angle. The centre of rotation is the point used in the G79 block to program the location of the pocket or groove.

The angle is programmed with the word B1= in degrees and decimal parts thereof and ranges from - 360 to 360'.

The angle is measured with the X-axis (G17 and G18) or the -Z-axis (G19).

If B1=0 is programmed or B1= is not programmed at all, the pocket or groove are milled axis parallel.

Note: The word B1= has two meanings in a G79 block. Either it is the angle for rotating a pocket or groove or it is used to program the coordinates (B1=, L1= or X/Y, B1=) for the position of the centre of the pocket.

GROUP 'A' FUNCTION (G0 to G3) AND G6

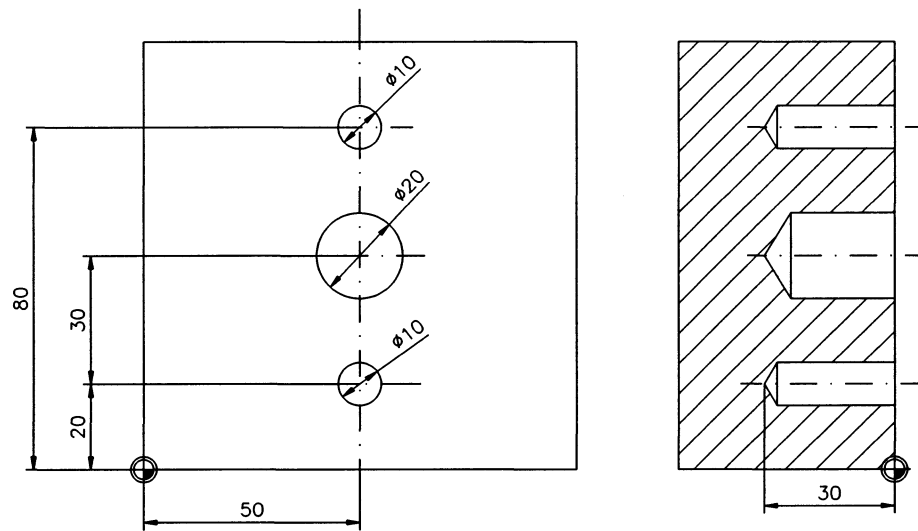
A group 'A' function is ignored by a G79-block. When the G79-block execution is finished, the group 'A' function will become active again.

TOOL POINTING IN POSITIVE DIRECTION (G66/G67)

If the tool is pointing in the positive direction of the tool axis (G67 activated), the depth of the fixed cycle must be programmed with a positive sign (+) to indicate that the cycle is to be executed in the positive direction of the tool axis.

With the milling cycles the direction of rotation on the circular arcs is automatically changed in the opposite direction.

The milling direction programmed with the J-word, is not automatically changed.

Examples**EXAMPLE 1. Three holes to be drilled**

```

N50 G78 P1 X50 Y20 Z0
N55 G78 P2 X50 Y80 Z0
N60 T1 M6
N65 G81 Y1 Z-30 F100 S1000 M3
N70 G79 P1=1 P2=2
N75 T2 M6
N80 G79 X50 Y50 Z0
N90 M30

```

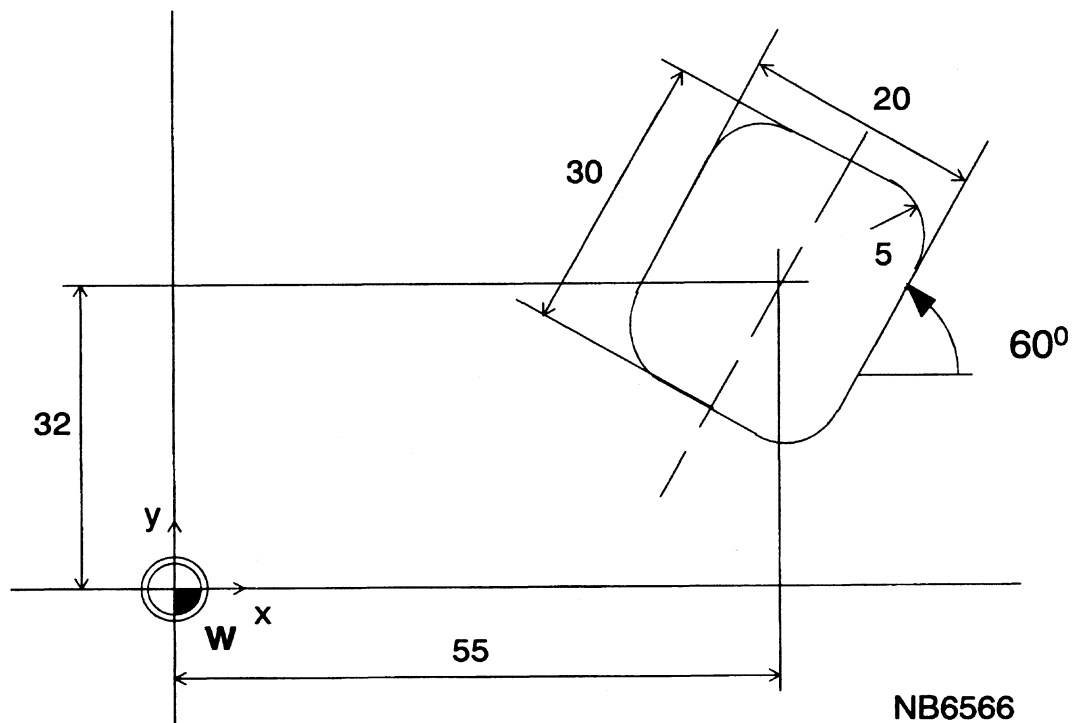
Explanation

```

N50: Define point 1
N55: Define point 2
N60: Load tool 1 (drill of diameter 10)
N65: Define drilling cycle and start the spindle
N70: Drill holes at points 1 and 2
N75: Load tool 2 (drill of diameter 20)
N80: Drill the hole
N90: End of program.

```

EXAMPLE 2. A rotated pocket



N55 G17
 N60 T1 M6
 N65 G87 X30 Y20 Z-5 B1 R5 F100 S1000 M3
 N70 G79 X55 Y32 Z0 B1=60

Explanation

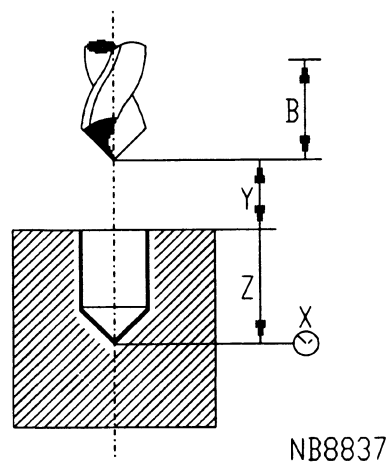
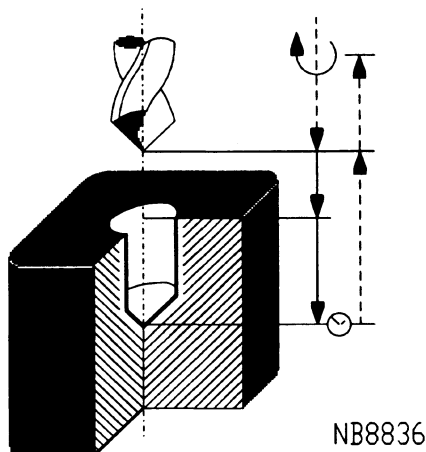
N55: Define the main plane for the pocket
 N60: Load tool 1, a mill with a radius of 4.5 mm
 N65: Define the pocket as if its sides are parallel to the X- and Y-axis
 N70: The pocket is milled. This block contains the centre of the pocket (X55, Y32, Z0) and the angle (60°) the axis of the pocket makes with the X-axis.

45. Drilling cycle G81

Purpose

To define in one program block the drilling of a hole.

Format



..... Rapid movement
 ----- Feed movement

N... G81 Y... Z... {X...} {B...}

Parameters

Y Clearance
 Z Drilling depth
 X Dwell time (sec)
 B Retract distance

Modal parameters
 F, S, T, M, H, E... =
 T1=, T2=

Associated functions

G77, G79, G83-G89

Type of function

Modal

Notes and usage

EXECUTION

A defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block.

The cycle is executed in the tool axis which is stated with the function for plane selection (G17, G18 or G19).

DEPTH OF OPERATION (Z)

Final depth of operation measured from the surface.

The sign of the Z-word indicates the direction of depth movement in the tool axis:

"-" in the negative direction, in most cases into the hole

"+" in the positive direction.

DWELL AT BOTTOM OF HOLE (X)

If required, a dwell at the bottom of the hole can be programmed in steps of .1 second.

Minimum programmable dwell period: .1 second

Maximum programmable dwell period: 900 seconds

If the X-word is not programmed, no dwell is executed.

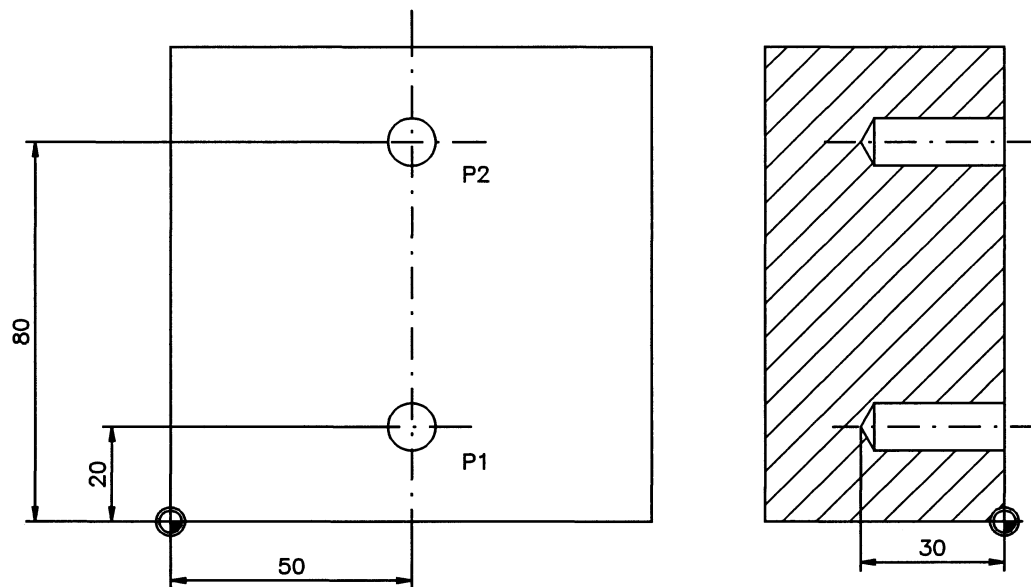
RETRACT DISTANCE (B)

The retract distance is added to the clearance value (Y-word). It can be used eg. to avoid obstacles. This extra distance can have either a positive or negative value.

If the B-word is not programmed, the retract movement is executed to a point the clearance distance above the surface.

Cancellation

The cycle's values are cancelled when a new cycle is defined or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

Example

```

N50 G78 P1 X50 Y20 Z0
N55 G78 P2 X50 Y80 Z0
N60 G0 Z10 T1 M6
N65 G81 X1.5 Y1 Z-30 F100 S500 M3
N70 G79 P1 P2

```

Explanation

N50: Define point 1
 N55: Define point 2
 N60: Load tool 1 and move tool out the tool change position
 N65: Define fixed drilling cycle and start the spindle
 N70: Execute fixed cycle at point 1 and then point 2.

Fixed cycle sequence:

drill moves at rapid traverse rate to a point the clearance distance (Y-word of G81) above the surface.

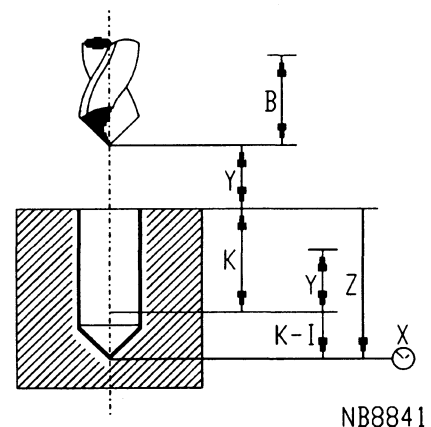
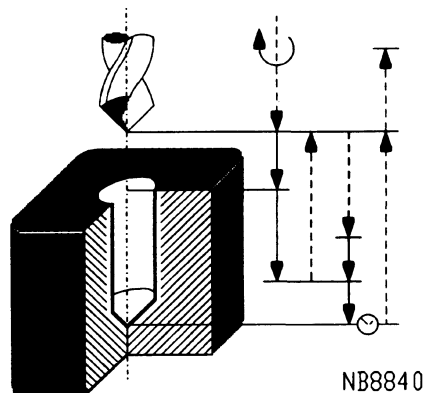
- drill feeds to depth (Z-word of G81) at set feedrate
- drill is stopped for 1.5 seconds
- drill is retracted at rapid traverse rate to a point the clearance distance above the surface.

46. Deep hole drilling cycle G83

Purpose

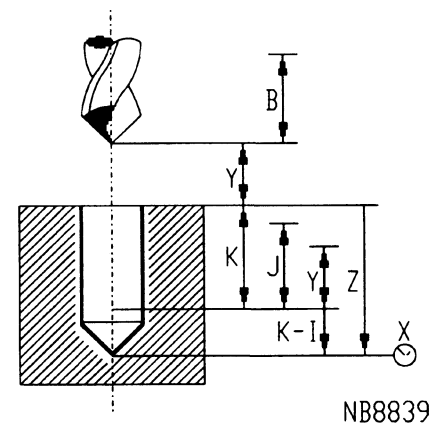
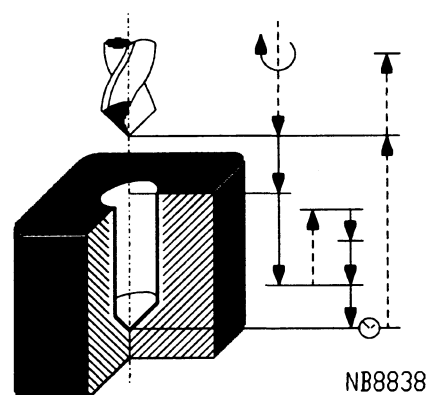
To define in one program block the drilling of a deep hole.

Format



Tool movements when J0 is active (tool is fully retracted to clearance position Y)

N... G83 Y... Z... K... J0 {I...} {X...} {B...}



Tool movements when J>0 (tool remains inside workpiece between cutting passes)

N... G83 Y... Z... K... J... {I...} {X...} {B...} {K1...}

Parameters

Y Clearance
Z Overall drilling depth
X Dwell time (sec)
B Retract distance
I Drilling depth decrement
J Retract distance after step
K Drilling depth first movement
K1= Number of retract distances
Modal parameters
F, S, T, M, H, E...=
T1=, T2=

Associated functions

G77, G79, G81, G84-G89

Type of function

Modal.

Notes and usage**EXECUTION**

A defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block.

The cycle is executed in the tool axis which is stated with the function for plane selection (G17, G18 or G19).

DEPTH OF OPERATION (Z)

Final depth of operation measured from the surface. The sign of the Z-word indicates the direction of depth movement in the tool axis:

"-" in the negative direction, in most cases into the hole
"+" in the positive direction.

DWELL AT BOTTOM OF HOLE (X)

If required, a dwell at the bottom of the hole can be programmed in steps of .1 second.

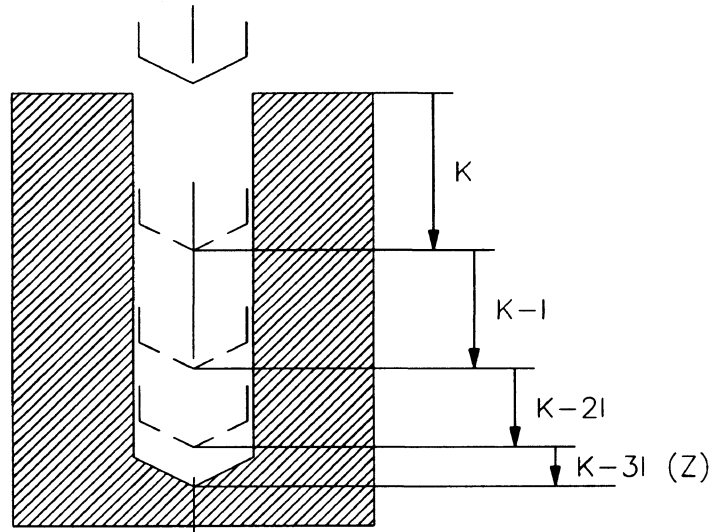
Minimum programmable dwell period: .1 second
Maximum programmable dwell period: 900 seconds
If the X-word is not programmed, no dwell is executed.

RETRACT DISTANCE (B)

The retract distance is added to the clearance value (Y-word). It can be used eg. to avoid obstacles. This extra distance can have either a positive or negative value.

If the B-word is not programmed, the retract movement is executed to a point the clearance distance above the surface.

INCREMENTAL REDUCTION (I)



NB8854a

If the calculated feed-in distance becomes less than the I-value, the I-value is used instead. The final feed-in distance can be smaller than the I-value.

The I-word has no sign.

I0: all feed-in distances (except perhaps the final one) are equal to the first feed-in distance (K-word).

RETRACT INDICATOR (J)

A special word (J) is used to indicate how the tool is to be retracted after each cutting pass:

J0: after each cutting step, the tool is retracted out of the hole to the point defined by the clearance distance.

J>0: after each cutting step, the tool is retracted over the programmed distance. In this way chips are broken, but the tool remains in the hole.

The J-word has no sign.

FIRST FEED-IN DISTANCE (K)

In general a deep hole drilling operation takes place in several steps. The drilling depth of the first step is programmed with the K-word. If the K-value is greater than the total depth (Z-word), the hole is drilled at depth in one cutting pass.

The K-word has no sign.

NUMBER OF SPECIAL RETRACT DISTANCE (K1=)

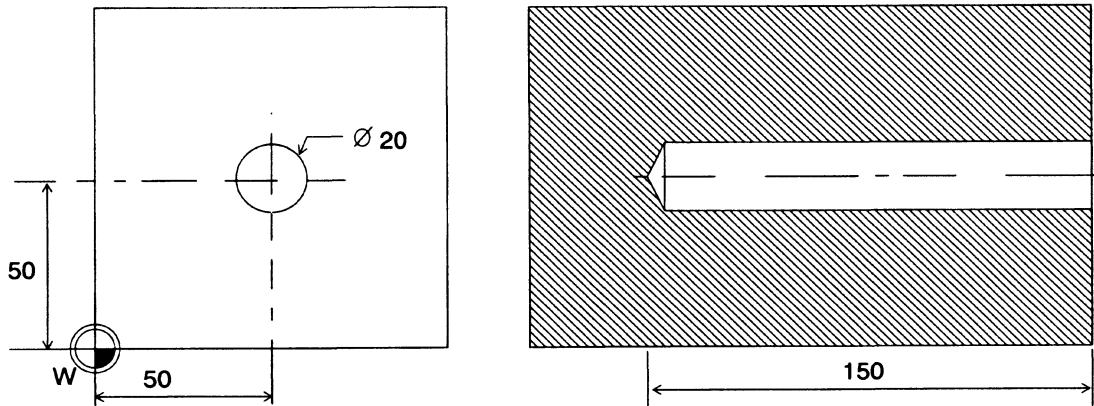
After a programmable number of feed movements defined by (K1 =), a retract will take place to a position the clearance distance before the preceding depth, to remove the chips. This only will take place if the special retract distance (J>0) is defined.

CANCELLATION

The cycles values are cancelled when a new cycle is defined or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

Example

EXAMPLE 1:



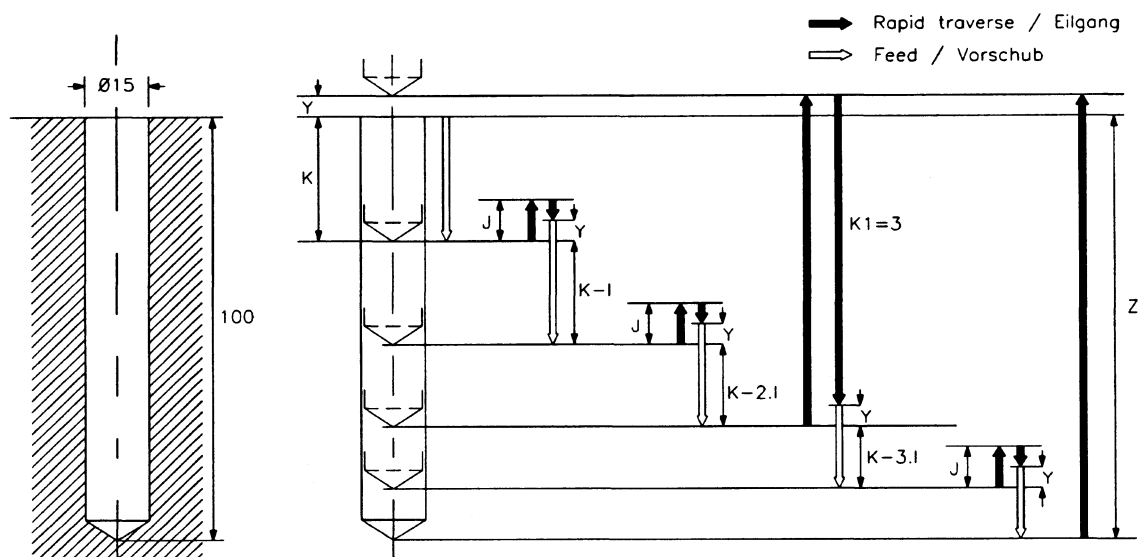
NB8553

```
N5 T1 M6
N10 G83 X3 Y4 Z-150 I2 J6 K20 F200 S500 M3
N20 G79 X50 Y50 Z0
```

N5: Load tool 1.
 N10: Define the fixed cycle for deep hole drilling.
 N20: Execute the fixed cycle at the programmed position

EXAMPLE 2:

Make a deep hole drilling (d=15, depth 100). After 3 feed movements a retract must take place.



NB9884

N.. G83 Y3 Z-100 I5 J6 K30 K1=3

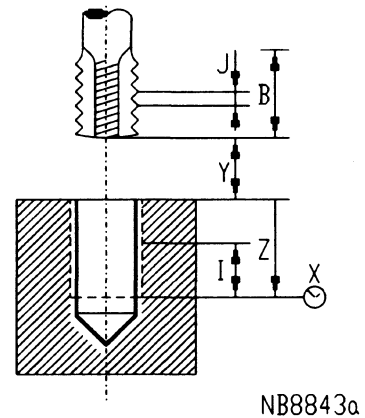
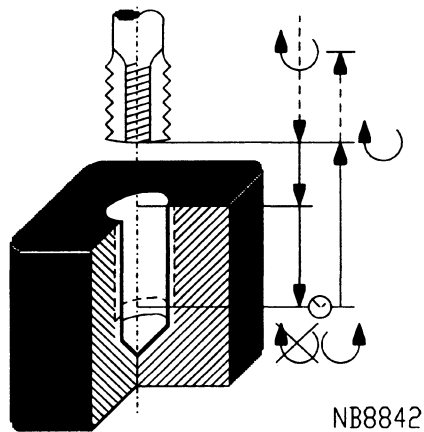
- G83 Deep hole drilling cycle
- Y3 Clearance distance 3mm
- Z-100 Depth of operation 100mm
- I5 Incremental reduction 5mm
- K30 First feed-in distance 30mm. K-Value is reduced with the I-Value every cutting pass.
- K1=3 After 3 feed movements a retract will take place to the clearance position.

47. Tapping cycle G84

Purpose

To define in one program block the tapping of a hole.

Format



N... G84 Z... Y... {B...} {I...} {J...} {X...}

Parameters

Y Clearance
Z Tapping depth
X Dwell time (sec)
B Retract distance
I Speed ramp (rev.)
J Pitch

Modal parameters

F, S, T, M, H, E... =
T1=, T2=,

Associated functions

G77, G79, G81, G83, G85-G89

Type of function

Modal

Notes and usage**EXECUTION**

A defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block. The cycle is executed in the tool axis which is stated with the function for plane selection (G17, G18 or G19).

FLOATING TAP HOLDER

When a floating tap holder is used, the clearance distance (Y) must be sufficient for the tool not to touch the workpiece when the tool is fully retracted and the tap holder spring is no longer under compression.

TAPPING DEPTH (Z)

Final tapping depth measured from the surface. The sign of the Z-word indicates the direction of the movement in the tool axis:

DWELL AT BOTTOM OF HOLE (X)

If required, a dwell at the bottom of the hole can be programmed in steps of 0.1 seconds.

Minimum programmable dwell period: .1 second

Maximum programmable dwell period: 900 seconds

If the X-word is not programmed, no dwell is executed.

RETRACT DISTANCE (B)

The retract distance is added to the clearance value (Y-word). It can be used eg. to avoid obstacles. This extra distance can have either a positive or negative value.

If the B-word is not programmed, the retract movement is executed to a point the clearance distance above the surface.

SPEED RAMP (REV.) (I)

This word is used by the CNC to determine the point where the spindle starts to safely slow down before the end of the thread is reached and stop at the bottom of the thread.

With the I-address the number of revolutions required for the spindle to safely slow down and stop at the bottom of the thread is programmed. If the I-word is not programmed, the CNC uses a Machine Constant value for establishing this point.

PITCH OF THE THREAD (J)

The pitch of the thread can be programmed:

- by using the J-word

- by programming the F(feedrate) = pitch(J) * spindle speed(S)

CANCELLATION

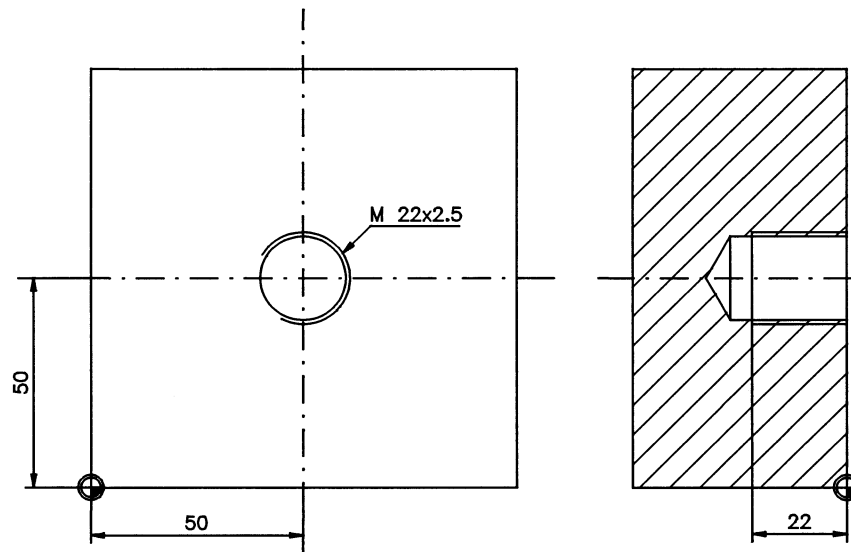
The cycle's values are cancelled when a new cycle is defined or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

RIGID TAPPING

If a transducer is mounted at the spindle of the machine tool, rigid tapping, thus tapping without using a floating tapholder, is possible.

To eliminate the drift of the spindle an oriented spindle stop (M19) must be programmed before the tapping starts. On some machine tools the oriented spindle stop is automatically executed with a tool change (M6). Refer to the machine tool builder's documentation for details.

Note: In case of G84 execution by G79 the control must be in G94 mode (F in mm/min) and not in G95 mode (F in mm/rev). The programmer should also always program G94 before G84.

Example**EXAMPLE 1.**

```
N14 T3 M6
N15 G84 Y9 Z-22 J2.5 S56 M3 F140
N20 G79 X50 Y50 Z0
```

Explanation:

- N14: Load tool 3 (Tap M22 x 2.5).
- N15: Define tapping cycle and start the spindle.
- N20: Execute the tapping cycle on the programmed position. A floating tap holder is used.

Tapping cycle sequence:

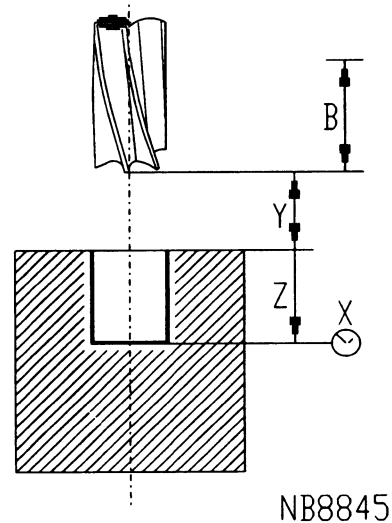
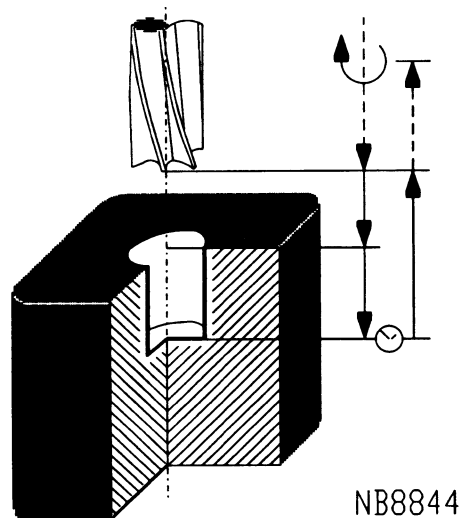
- tap moves at rapid travers rate to a point the clearance distance (Y-Word) above the surface. Spindle rotation clockwise (M3).
- tap feeds to tapping depth at a set feedrate. (the feed depends on pitch and spindle speed).
- retract the tap to a point the clearance distance above the surface after the spindle rotation is switched to counter-clockwise.

48. Reaming cycle G85

Purpose

To define in one program block the reaming of a hole.

Format



N... G85 Z... Y... {B...} {X...} {F3=..}

Parameters

Y Clearance
Z Reaming depth
X Dwell time (sec)
B Retract distance
F2= Feedrate to startpoint

Modal parameters

F,S, T,M,H,E...=

T1=, T2=

Associated functions

G77, G79, G81, G83, G84, G86-G89

Type of function

Modal

Notes and usage**EXECUTION**

A defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block. The cycle is executed in the tool axis which is stated with the function for plane selection (G17, G18 or G19).

REAMING DEPTH (Z)

Reaming depth measured from the surface.

The sign of the Z-word indicates the direction of depth movement in the tool axis:

"-" in the negative direction, in most cases into the hole

"+" in the positive direction.

DWELL AT BOTTOM OF HOLE (X)

If required, a dwell at the bottom of the hole can be programmed in steps of .1 second.

Minimum programmable dwell period: .1 second

Maximum programmable dwell period: 900 seconds

If the X-word is not programmed, no dwell is executed.

RETRACT DISTANCE (B)

The retract distance is added to the clearance value (Y-word). It can be used eg. to avoid obstacles. This extra distance can have either a positive or negative value.

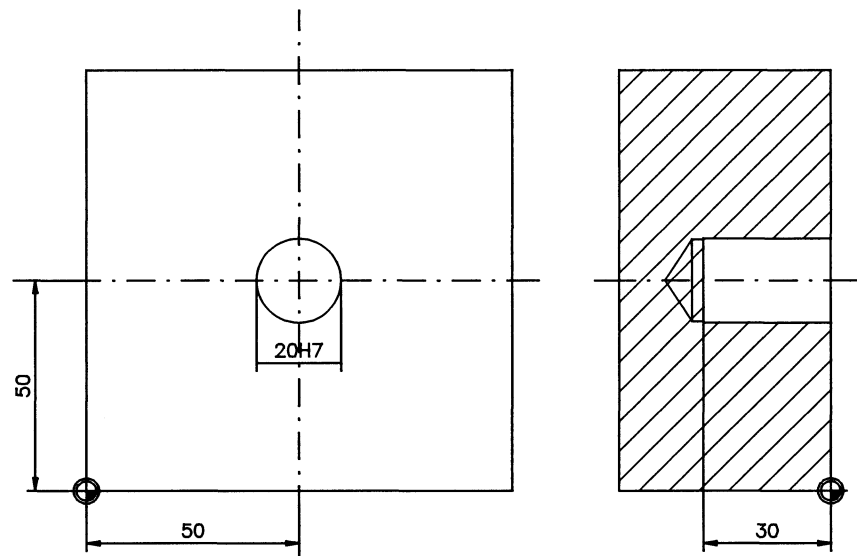
If the B-word is not programmed, the retract movement is executed to a point the clearance distance above the surface.

FEEDRATE TO STARTPOINT (F2=)

Feedrate to startpoint (F2=) is the feed from depth to starting position. This Feed (F2=) allows a faster retract feed and can be programmed. This will reduce the overall cycle execution duration. If the Feed F2= is not defined in a cycle the Feed to the starting position, is the programmed Feed (F).

CANCELLATION

The cycles values are cancelled when a new cycle is defined or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

Example

```

N25 Z10 T4 M6
N30 G85 X2 Y3 Z-20 F50 S100 M3
N35 G79 X50 Y50 Z0 F2=200

```

Explanation:

N25: Load tool 4, the reamer
 N30: Define the reaming cycle and start the spindle
 N35: Execute the reaming cycle at the programmed position

Cycle sequence:

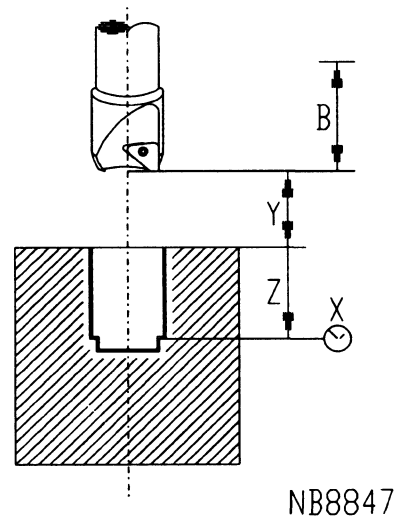
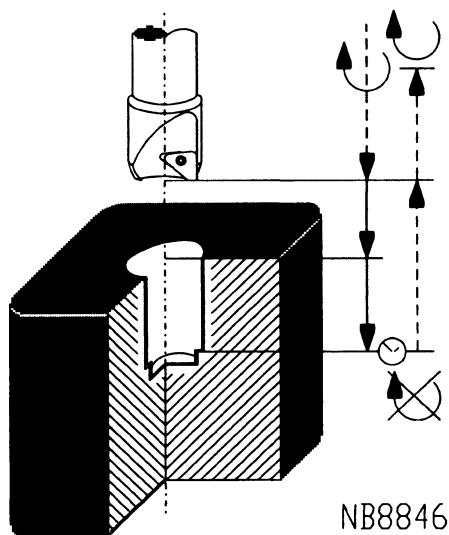
- the reamer moves at rapid traverse rate to the clearance distance (Y3)
- the reamer feeds to depth (Z-20) at feedrate (F50)
- a dwell of 2 seconds (X2) at the bottom of the hole
- the reamer is retracted at feedrate (F2=200) to the clearance distance (Y3)

49. Boring cycle G86

Purpose

To define in one program block the boring of a hole.

Format



N... G86 Z... Y... {X...} {B...}

Parameters

Y Clearance
Z Boring depth
X Dwell time (sec)
B Retract distance

Modal parameters

F, S, T, M, H, E... =

T1=, T2=

Associated functions

G77, G79, G81, G83-G85, G87-G89

Type of function

Modal

Notes and usage**EXECUTION**

A defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block.

The cycle is executed in the tool axis which is stated with the function for plane selection (G17, G18 or G19).

BORING DEPTH (Z)

Boring depth measured from the surface.

The sign of the Z-word indicates the direction of depth movement in the tool axis:

"-" in the negative direction, in most cases into the hole

"+" in the positive direction.

DWELL AT BOTTOM OF HOLE (X)

If required, a dwell at the bottom of the hole can be programmed in steps of .1 second.

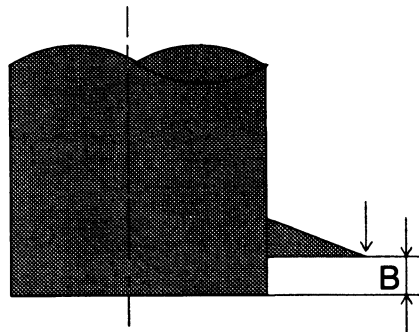
Minimum programmable dwell period: .1 second

Maximum programmable dwell period: 900 seconds

If the X-word is not programmed, no dwell is executed.

RETRACT DISTANCE (B)

The retract distance is added to the clearance value (Y-word). It can be used to compensate for a bore whose tool tip is not on the bottom of the boring bar.



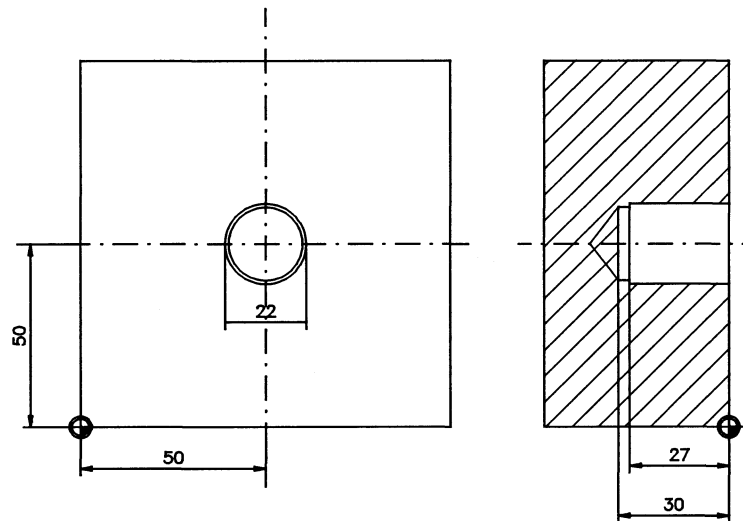
NB8551

If this extra distance is not added, the bottom of the boring bar could still be inside the workpiece after the tool tip has been retracted the clearance distance.

If the B-word is not programmed, the tool tip is retracted to the clearance distance above the surface.

CANCELLATION

The cycles values are cancelled when a new cycle is defined or by soft key CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

Example**EXAMPLE 1.**

N45 T5 M6
 N50 G86 X1 Y9 Z-27 B10 F100 S500 M3
 N55 G79 X50 Y50 Z0

Explanation:

N45: Load tool 5, a boring bar.
 N50: Define boring cycle and start the spindle.
 N55: Execute fixed cycle at the programmed position

Fixed cycle sequence:

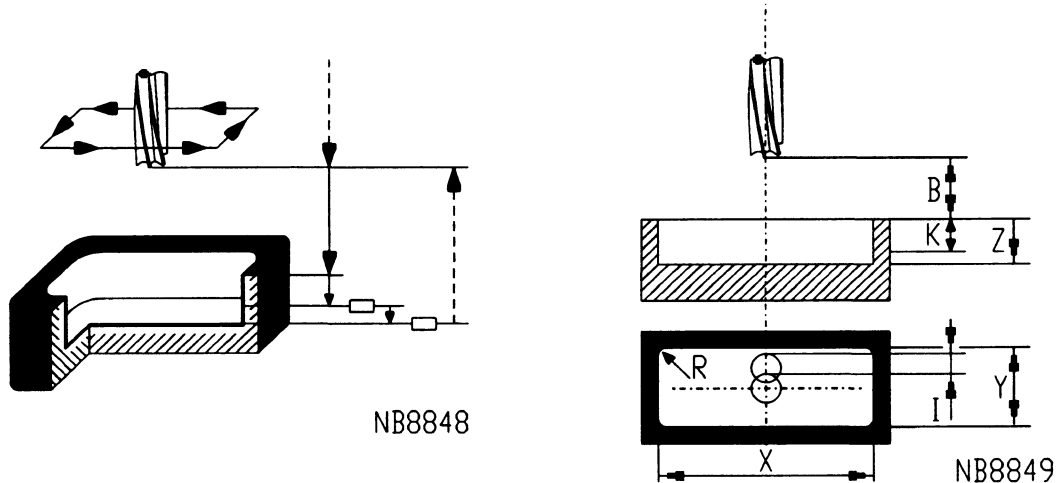
- the tool tip of the bore on the boring bar moves at rapid traverse rate to the clearance distance (Y-word).
- bore feeds to depth (Z-word) at feedrate
- at depth the spindle is stopped
- a dwell of 1 second
- the tool tip of the bore is retracted at rapid traverse rate to the clearance distance(Y-word).
- The tool tip is out of the hole.
- the spindle is started again
- retract the tool tip to a point the retract distance (B-word) above the surface.

50. Rectangular pocket milling cycle G87

Purpose

To define in one program block the geometry of a rectangular pocket and some parameters for cutting the pocket.

Format



N... G87 X... Y... Z... B... R... {J...} {I...} {K...} {Y3=...} {F2=...}

Parameters

Pocket geometry

X Dimension parallel to X

Y Dimension parallel to Y

Z Total pocket depth

R Corner radius

F2= In depth feed (only this block)

Cutting parameters

B Clearance

I Cutting width mill in %

J J1:climb / J-1:conventional

K Cutting depth

Y3= Special retract distance

Modal parameters

F,S, T,M,H, E...=

T1=, T2=

Associated functions

G77, G79, G81, G83-G86, G88, G89

Type of function

Modal

Notes and usage

EXECUTION

A defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block. In these blocks the coordinates of the centre of the pocket are stated.

The pocket is milled in the active plane (G17, G18 or G19). The radius of the actual tool stored in the tool memory, is used for milling the pocket.

TOTAL POCKET DEPTH (Z)

Total pocket depth measured from the surface.

The sign of the Z-word indicates the direction of depth movement in the tool axis:

"-" in the negative direction, in most cases into the hole

"+" in the positive direction.

PLANE OF OPERATION

The table below lists which axes are involved with the words X and Y to define the pocket dimensions in each of the three main planes.

	XY-PLANE	XZ-PLANE	YZ- PLANE
X-word parallel to	X-axis	X-axis	Z-axis
Y-word parallel to	Y-axis	Z-axis	Y-axis
Z-word (tool axis)	Z-axis	Y-axis	X-axis

The X and Y-word are programmed without sign.

FEEDRATE TO CUTTING DEPTH (F2=)

Feedrate to cutting depth (F2=) is the feed of the depth movement in the Toolaxis for each cleaning pass.

If the Feed F2= is not defined in a cycle the Feed to the depth (K-word), is half the programmed feed (F).

CUTTING WIDTH IN % (I)

The value of the I-word states the percentage of the tool diameter to be used as cutting width for each cutting pass.

For example: I75 states that the cutting width is equal to 75% of the tool diameter.

If the I-word is not programmed, the value of a machine constant(MC720) is used.

CUTTING DEPTH (K)

If the pocket can not be cleaned out at depth in one pass, the K-word is used to program the depth for each cleaning pass.

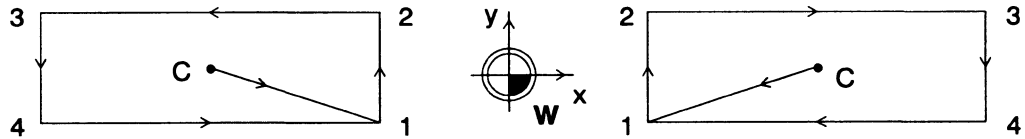
If the K-word is not programmed or has the same value as the Z-word, the pocket is milled in one pass.

SPECIAL RETRACT DISTANCE (Y3=)

The special retract distance Y3= is added to the clearance distance (B-word) this gives the opportunity to avoid objects. The special retract distance can have a sign, which will normally be the positive one.

If the special retract distance (Y3=) is not defined a retract movement is executed to the clearance distance.

DIRECTION OF MILLING (J)



NB6075

J+1 Climb milling

J-1 Conventional milling

If no direction of milling is programmed, climb milling is assumed (J1 = default direction).

ROTATED POCKET

If the pocket is to be milled making an angle with the main axes, the G77 and G79 block are extended with a special word (B1=) to indicate the angle of rotation. Refer to the functions G77 and G79 ROTATED POCKET OR GROOVE for additional information.

Note: In the cycle the pocket is defined as if it is parallel to the axes.

CANCELLATION

The cycle's values are cancelled when a new cycle is defined or by softkey CLEAR CONTROL, M30 or by CANCEL PROGRAM.

FINISHING THE POCKET

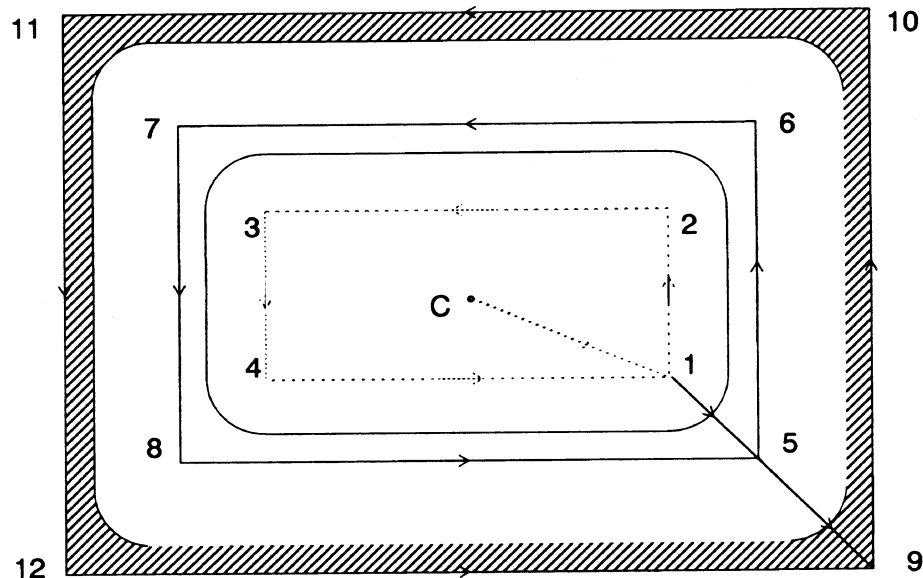
If a pocket must be milled and must have a finishing pass the following method may be used:

1. Add the allowance of the finishing cut to the tool radius and store the larger radius in the tool memory.
2. Execute the G87 milling cycle. A contour is milled which is smaller than the programmed contour due to the difference between the tool memory radius and the actual radius.
3. Program the contour of the pocket with the regular G1 and G2/G3 functions and execute the finishing pass with radius compensation and use the actual tool diameter.

THE TOOL SEQUENCE

The tool sequence for milling the pocket is:

- With rapid traverse to the centre (C) of the pocket and stay the clearance distance (B-word) above the workpiece.
- With half the programmed feed to the first depth (K-word).



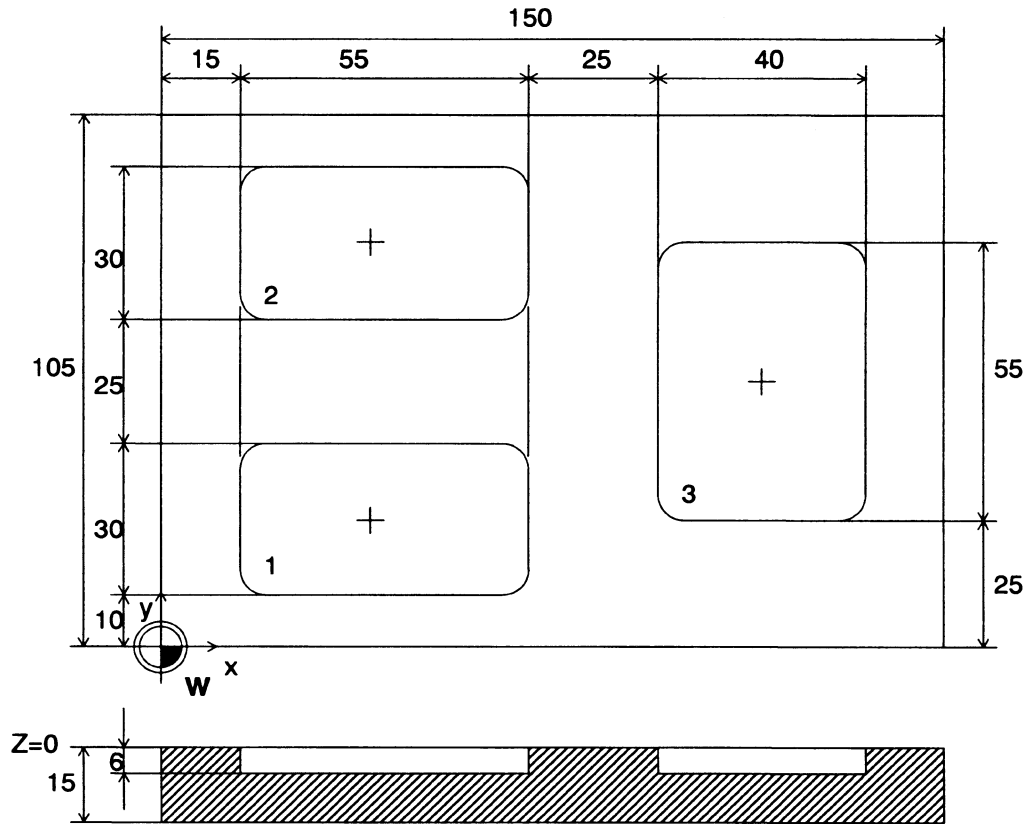
NB6076

Pocket milling sequence

- Move the tool from the centre to point 1 and mill around from 1 to 2, 3, 4 and back to 1. Point 1 is calculated by the control and depends on the X-word, Y-word and the radius of the active tool.
- Move the tool to point 5. The point is calculated by the control. The distances parallel to the axes are: I-word x tooldiameter
- Move the tool around from 5 to 6, 7, 8 and back to 5.
- Repeat the steps d and e -if necessary- until the layer is cleaned out.
- Finally follow the programmed contour and stop in the centre of the corner.
- If the programmed depth is reached, retract the tool to the clearance.
- If the programmed depth is not reached, move, with three times the programmed feed, to the centre (C) of the pocket.
- Clean out another layer by repeating the steps b to i.

After the cleaning out, a finishing for the sides of the pocket might be necessary. The best way to proceed is to store in the tool memory, for the actual tool, a radius being the stock removal greater than the actual radius of the tool. Once the cycle is totally executed this stock removal remains for finishing. The finishing of the pocket is programmed by activating the tool radius compensation and using the regular G1 and G2/G3 blocks.

Milling three pockets.



NB5810

Example with three pockets.

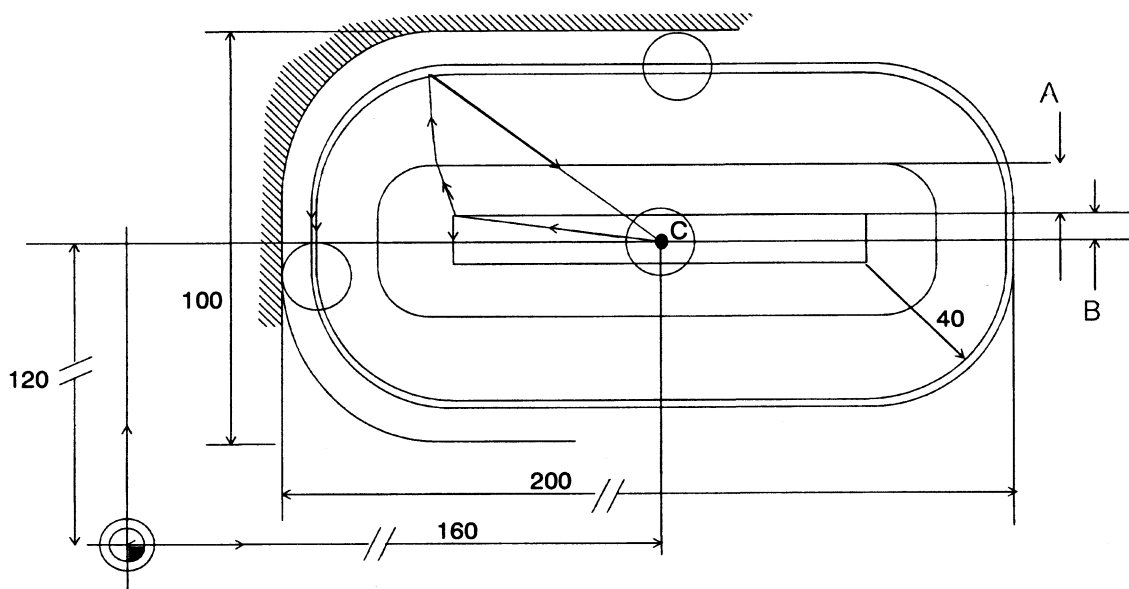
```

N10 T1 M6 (Mill R5)
N20 G87 X55 Y30 Z-6 J+1 B1 I75 K1.5 F200 S500 M3
N30 G79 X42.5 Y25 Z0
N31 G79 X42.5 Y80 Z0
N40 G87 X40 Y55 Z-6 J+1 B1 I75 K1.5 F200 S500 M3
N50 G79 X115 Y52.5 Z0

```

Example

EXAMPLE 1



NB8639

A: 75% of the diameter

B: 75% of the radius

N10 T1 M6 (Mill R5)

N20 G87 X200 Y100 Z-6 J+1 B1 R40 I75 K1.5 F200 S500 M3

N30 G79 X160 Y120 Z0

Explanation:

N10: Load tool 1.

N20: Define pocket milling cycle.

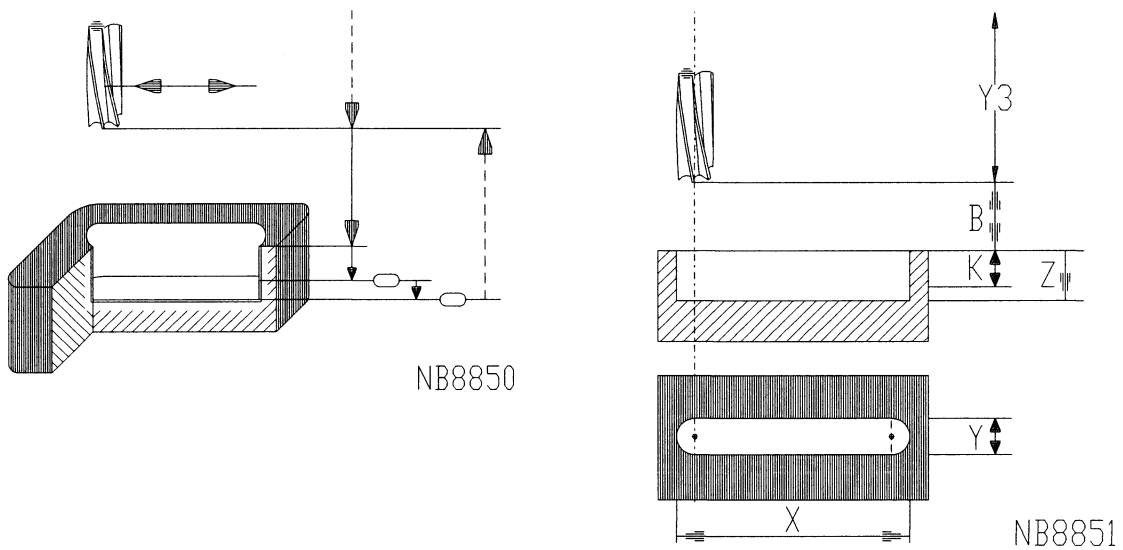
N30: Execute cycle at programmed position.

51. Groove milling cycle G88

Purpose

To define in one program block the geometry of a groove and some parameters for milling it.

Format



N... G88 X... Y... Z... B... {J...} {K...} {Y3=...} {F2=...}

Parameters

Groove geometry

X Dimension parallel to X

Y Dimension parallel to Y

Z Total groove depth

F2= In depth feed (only this block)

Cutting parameters

B Clearance

J J1:climb / J-1:conventional

K Cutting depth

Y3= Special retract distance

Modal parameters

F,S,T,M,H,E...=

T1=, T2=

Associated functions

G77, G79, G81, G83-G87, G89

Type of function

Modal

Notes and usage

EXECUTION

defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block. In these blocks the coordinates of the entry point of the groove are stated.

The groove is milled in the active plane (G17, G18 or G19). The radius of the actual tool stored in the tool memory, is used for milling the groove.

TOTAL DEPTH (Z)

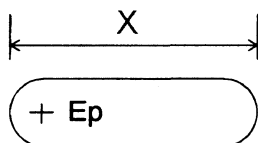
Total depth of the groove measured from the surface.

The sign of the Z-word indicates the direction of depth movement in the tool axis:

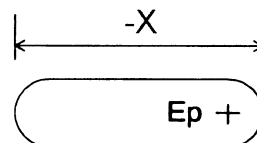
"-" in the negative direction, in most cases into the hole

"+" in the positive direction.

GROOVE PARALLEL TO X-AXIS



X... (G17), X... (G18), Y... (G19)



X... (G17), X... (G18), Y... (G19)

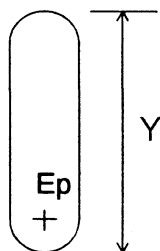
(NB8683)

If the axis of the groove should be parallel to the X-axis, the programming is as follows:

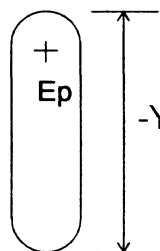
absolute value of X-word > than value of Y-word

The sign of the X-word determines on which side of the entry point (Ep) the groove is milled. The Y-word is programmed without sign.

GROOVE PARALLEL TO Y-AXIS



Y... (G17), Y... (G18), X... (G19)



Y... (G17), Y... (G18), X... (G19)

(NB8683)

If the axis of the groove should be parallel to the Y-axis, the programming is as follows:

absolute value of Y-word > than value of X-word

The sign of the Y-word determines on which side of the entry point (Ep) the groove is milled. The X-word is programmed without sign.

PLANE OF OPERATION

The table below lists which axes are involved with the words X and Y to define the length and width of a groove in the three main planes.

	XY-PLANE	XZ-PLANE	YZ-PLANE
X-word parallel to	X-axis	X-axis	Z-axis
Y-word parallel to	Y-axis	Z-axis	Y-axis
Z-word (tool axis)	Z-axis	Y-axis	X-axis

FEEDRATE TO CUTTING DEPTH (F2=)

Feedrate to cutting depth (F2=) is the feed of the depth movement in the Toolaxis for each cleaning pass.

If the Feed F2= is not defined in a cycle the Feed to the depth (K-word), is half the programmed feed (F).

CUTTING DEPTH (K)

If the groove can not be cleaned out at depth in one pass, the K-word is used to program the depth for each cleaning pass.

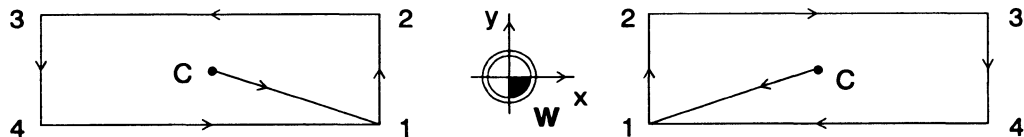
If the K-word is not programmed or has the same value as the Z-word, the groove is milled in one pass.

SPECIAL RETRACT DISTANCE (Y3=)

The special retract distance Y3= is added to the clearance distance (B-word) this gives the opportunity to avoid objects. The special retract distance can have a sign, which will normally be the positive one.

If the special retract distance (Y3=) is not defined a retract movement is executed to the clearance distance.

DIRECTION OF MILLING ON THE FINISHING PATH (J)



NB6075

J+1 Climb milling

J-1 Conventional milling

If no direction of milling is programmed, climb milling is assumed (J1 = default direction).

ROTATED GROOVE

If the groove is to be milled making an angle with the main axes, the G77 and G79 block are extended with a special word (B1=) to indicate the angle of rotation. Refer to the functions G77 and G79 ROTATED POCKET OR GROOVE for additional information.

Note: In the cycle the groove is defined as if it is parallel to the axes.

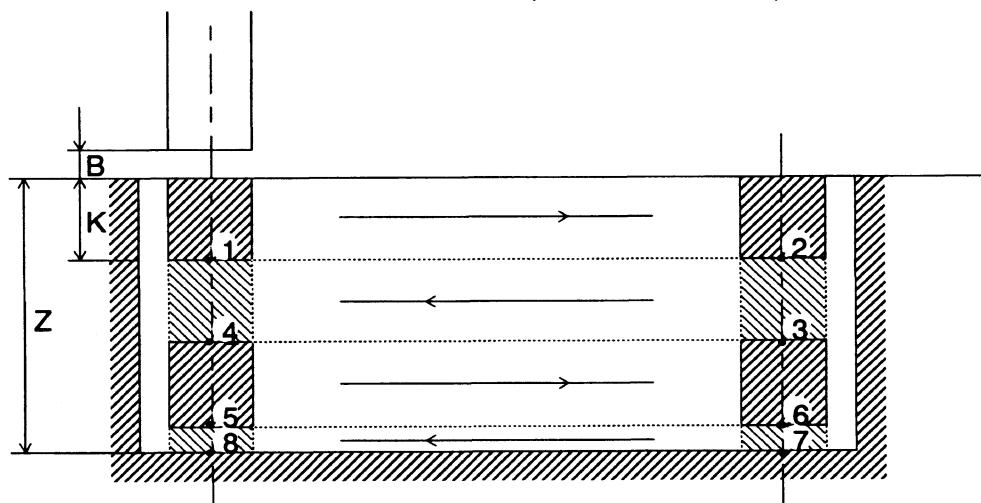
CANCELLATION

The cycle's values are cancelled when a new cycle is defined or by softkey CLEAR CONTROL, M30 or by CANCEL PROGRAM.

THE TOOL SEQUENCE

The tool sequence for milling the groove is:

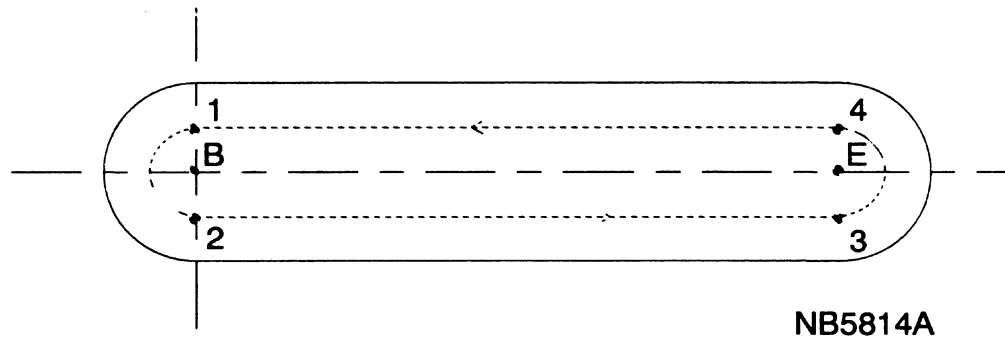
- With rapid to point B where the tool enters the groove and stay the clearance distance (B-word) above the workpiece.
- With half the programmed feed to the first depth (1).
- With the programmed feed through the centre of the groove to point E (2).
- With half the programmed feed to the second depth (3).
- With the programmed feed through the centre of the groove back to point B (4).
- So the tool moves to and from, each time at another depth until the final depth is reached.



NB7977

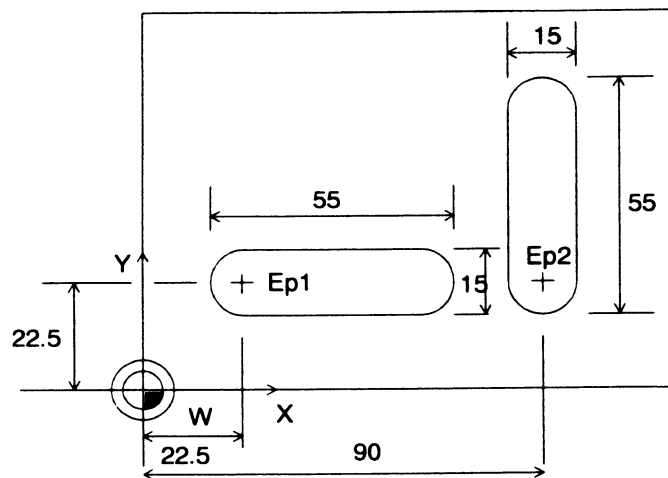
Depth movements in the groove.

- Once the final depth is reached, the sides of the groove are milled from B to 1, 2, 3, 4, 1, and back to B in a counter clockwise direction or a clockwise direction if J-1 is programmed. Here the tool radius compensation is automatically activated by the control and cancelled when the cycle is completed. The radius of the tool is used with the radius compensation.



Cutter path for the sides.

- h. At the end of the cycle the tool is retracted out of the groove and stopped the clearance above the workpiece.

Example**EXAMPLE 1.**

NB8640

```

N10 T1 M6 (Mill R5)
N20 G88 X55 Y15 Z-5 B1 K1 Y3=10 F100 F2=200 S500 M3
N30 G79 X22.5 Y22.5 Z0
N40 G88 X15 Y55 Z-5 B1 K1 Y3=10 F2=200
N50 G79 X90 Y22.5 Z0

```

Explanation:

N10: Load tool 1.
 N20: Define groove milling cycle; parallel to X-axis.
 N30: Execute cycle at programmed position (Ep1).
 N40: Define groove milling cycle; parallel to Y-axis.
 N50: Execute cycle at programmed position (Ep2).

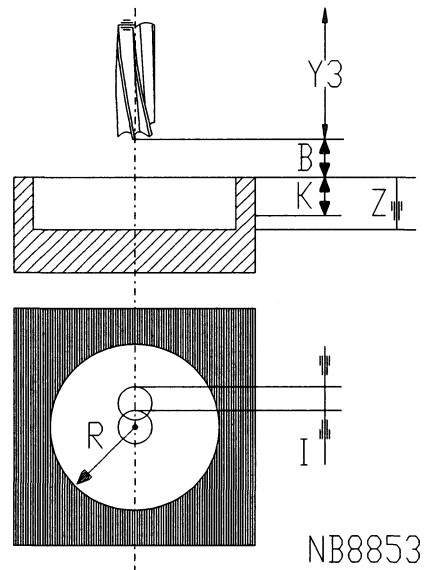
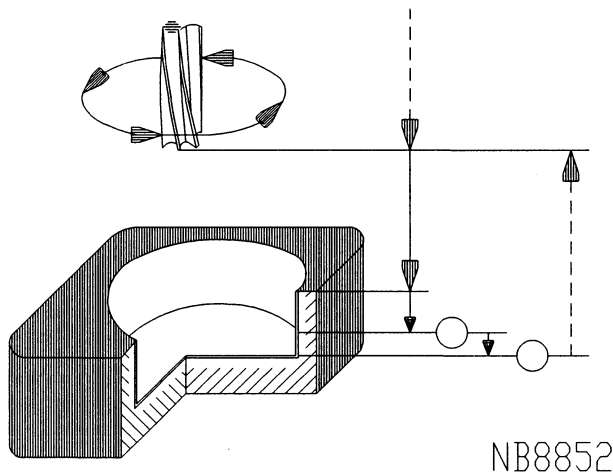
NOTE: The F, S and M functions are still active and therefore need not to be programmed again.

52. Circular pocket milling cycle G89

Purpose

To define in one program block the geometry of a circular pocket and some parameters for cutting the pocket.

Format



N... G89 Z... R... B... {1...} {J...} {K...} {Y3=...} {F2=...}

Parameters

Pocket geometry

Z Total pocket depth

R Radius circular pocket

F2= In depth feed (only this block)

Cutting parameters

B Clearance

I Cutting width mill in %

J J1:climb / J-1:conventional

K Cutting depth

Y3= Special retract distance

Modal parameters

F, S, T, M, H, E...=

T1=, T2=

Associated functions

G77, G79, G81, G83-G88

Type of function

Modal

Notes and usage

EXECUTION

A defined fixed cycle is executed on the position(s) programmed in either the G77 or G79 block. In these blocks the coordinates of the centre of the pocket are stated.

The pocket is milled in the active plane (G17, G18 or G19). The radius of the actual tool stored in the tool memory, is used for milling the pocket.

Fixed cycles should always be programmed with the G40-mode.

TOTAL POCKET DEPTH (Z)

Total pocket depth measured from the surface. The sign of the Z-word indicates the direction of depth movement in the tool axis:

"-" in the negative direction, in most cases into the hole

"+" in the positive direction.

PLANE OF OPERATION

The table below lists which axis in the active plane is used as the tool-AXIS.

	XY-PLANE	XZ-PLANE	YZ-PLANE
Z-word (tool axis)	Z-axis	Y-axis	X-axis

FEEDRATE TO CUTTING DEPTH (F2=)

Feedrate to cutting depth (F2=) is the feed of the depth movement in the Toolaxis for each cleaning pass.

If the Feed F2= is not defined in a cycle the Feed to the depth (K-word), is half the programmed feed (F).

CUTTING WIDTH IN % (I)

The value of the I-word states the percentage of the tool diameter to be used as cutting width for each cutting pass.

For example: I75 states that the cutting width is equal to 75% of the tool diameter.

If the I-word is not programmed, the value of a machine constant is (MC720).

CUTTING DEPTH (K)

If the pocket can not be cleaned out at depth in one pass, the K-word is used to program the depth for each cleaning pass.

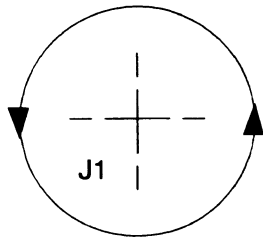
If the K-word is not programmed or has the same value as the Z-word, the pocket is milled in one pass.

SPECIAL RETRACT DISTANCE (Y3=)

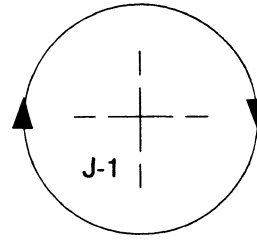
- The special retract distance Y3= is added to the clearance distance (B-word) this gives the opportunity to avoid objects.
- The special retract distance can have a sign, which will normally be the positive one.

If the special retract distance (Y3=) is not defined a retract movement is executed to the clearance distance.

DIRECTION OF MILLING (J)



J+1 Climb milling



J-1 Conventional milling

NB8554

If no direction of milling is programmed, climb milling is assumed (J1= default direction).

CANCELLATION

The cycle's values are cancelled when a new cycle is defined or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

FINISHING THE POCKET

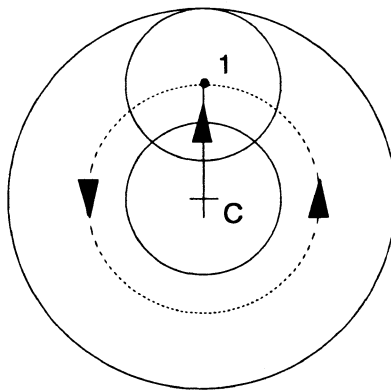
If a pocket must be milled and must have a finishing pass the following method may be used:

1. Add the allowance of the finishing cut to the tool radius and store the larger radius in the tool memory.
2. Execute the G89 milling cycle. A contour is milled which is smaller than the programmed contour due to the difference between the tool memory radius and the actual radius.
3. Program the contour of the pocket with the regular G1 and G2/G3 functions and execute the finishing pass with radius compensation and use the actual tool diameter.

TOOL SEQUENCE

The tool sequence for milling the circular pocket is:

- a. With rapid to the centre (C) of the pocket and stay the clearance distance (B-word) above the workpiece
- b. With half the programmed feed to the first depth (K-word)

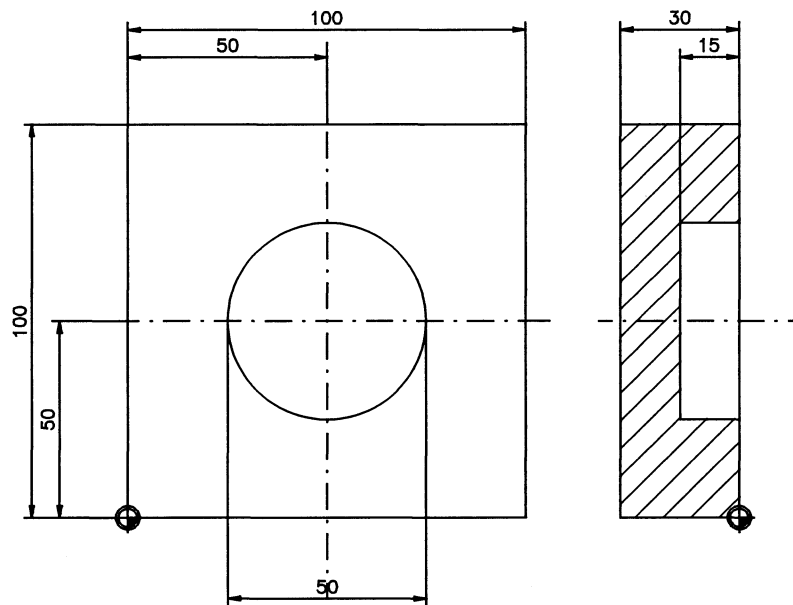


NB5817

Tool sequence for a circular pocket.

- c. With the programmed feed from C to 1. The distance to move is: $\text{tooldiameter} \times \text{I-word}$.
- d. Make with the programmed feed a full circle in clockwise direction (J-1) or counter clockwise direction (J+1) as seen from the tool.
- e. The steps c and d are repeated until all material is cleaned out from the first layer.
- f. Go with three times the programmed feed back to point C.
- g. If the programmed depth is not reached, another movement over the depth (K-word) takes place and then another layer is cleaned out.
- h. If the total depth is reached, retract the tool out of the pocket and stop the clearance above the workpiece.

After cleaning out, a finishing for the side of the pocket might be necessary. The best way to proceed is to store in the tool memory for the actual tool a radius, being the stock removal greater than the actual radius of the tool. Once the cycle is totally executed this stock removal remains for finishing. The finishing of the pocket is programmed by activating the tool radius compensation and using the regular G1 and G2/G3-blocks.

Example**EXAMPLE 1**

```
N10 T1 M6 (Mill R5)
N20 G89 Z-15 B1 R25 I75 K6 F200 S500 M3
N30 G79 X50 Y50 Z0
N40 G0 Z200
```

Explanation:

N10: Load tool 1.
N20: Define the cycle for milling a circular pocket.
N30: Execute cycle at programmed position.
N40: Retract the tool

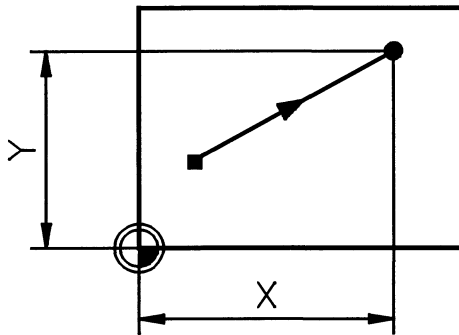
53. Absolute/Incremental programming G90/G91

Purpose

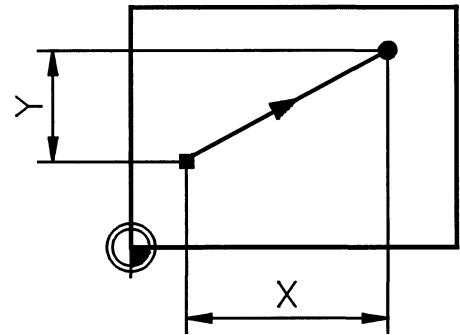
To select one of two modes of coordinate programming.

G90: Absolute coordinates measured from program zero point W.

G91: Incremental coordinates measured from last programmed tool position.



G90: Absolute coordinates



G91: Incremental coordinates

Format

N... G90/G91 [Axis coordinates]

Parameters

Axis coordinates with G90

X,Y,Z Endpoint coordinate

A,B,C Endpoint angle

Axis coordinates with G91

X,Y,Z Endpoint coordinate

A,B,C Endpoint angle

Type of function

Modal

Notes and usage

CANCELLATION

The G91 can be cancelled when G90 is defined or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

DEFAULT MODE

The G90 absolute coordinate mode automatically becomes active when the CNC system is switched on or at CLEAR CONTROL.

SWITCHING BETWEEN THE TWO MODES

The G90 default mode is cancelled by programming G91. The coordinates which follow the switch are interpreted by the CNC as incremental coordinates. To reactivate the absolute mode, G90 has to be programmed.

Internally the control operates with absolute coordinates. Therefore, within a particular program it is possible to change arbitrarily from absolute to incremental and vice versa.

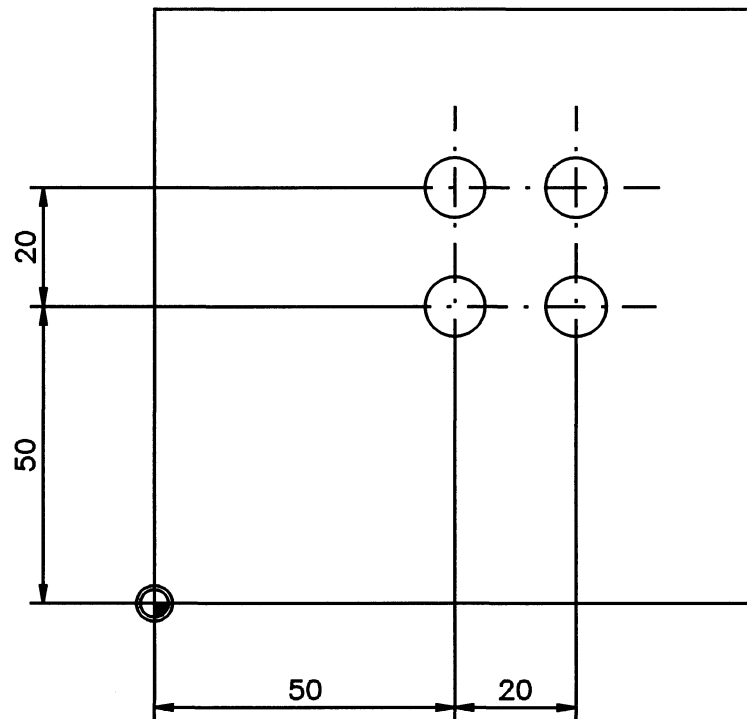
POLAR COORDINATES

The polar coordinates (B1=, L1=), (B2=, L2=), (B3=, L3=) are not influenced by the functions G90 and G91. They can be used arbitrarily in each coordinate mode.

POSITION DISPLAY

The axis positions on the display of the control are always absolute coordinates and related to the program zero point W.

Note: A part program should always contain an absolute position, before G91 is used. If a program starts immediately with G91, the actual tool position is used as the first absolute position of the program. Each time the program runs, this position should be the same, otherwise the program is executed each time at another place.

Example**EXAMPLE 1.**

```

N88550
N1 G17
N2 G54
N3 G195 X0 Y0 Z60 I100 J100 K-80
N4 S1300 T1 M6 (Drill R5)
N5 G0 X0 Y0 Z50
:
N8 G81 Y2 Z-10 F200 M3
N9 G79 X50 Y50 Z0
N10 G91
N11 G79 Y20
N12 G79 X20
N13 G79 Y-20
N14 G90
:
N17 G0 X0 Y0 Z50
N18 M30

```

Explanation:

```

N1 : Set the plane to be the XY-plane
N2 : Set the zero point
N3 : Graphic window definition
N4 : activate Tool 1
N5 : Move tool with rapid speed to start position
N8 : Drilling cycle definition
N9 : Drilling the first hole
N10 : Switching to incremental mode
N11 : A incremental movement to the second hole
N12 : A incremental movement to the third hole
N13 : A incremental movement to the third hole
N14 : Switching to absolute mode

```

N17 : Retract tool

N18 : End of program

54. Wordwise absolute and incremental programming (from V320)

Purpose

Wordwise absolute and incremental programming, independently of G90/G91.

Format

Programming absolute:

N.. G.. [Axisname]90=...

Programming incremental:

N.. G.. [Axisname]91=...

Parameters

Axisname:	X, Y, Z, U, V, W, I, J, K, A, B, C
X90=,Y90=,Z90=	Absolute endpoint
A90=,B90=,C90=	Absolute endpoint angle
X91=,Y91=,Z91=	Incremental endpoint
A91=,B91=,C91=	Incremental endpoint angle

Associated functions

G0, G1, G2, G3, G9, G45, G46, G61, G62, G77, G79, G145, G182

Type of function

Non modal

Notes and usage

Cartesian coordinates:

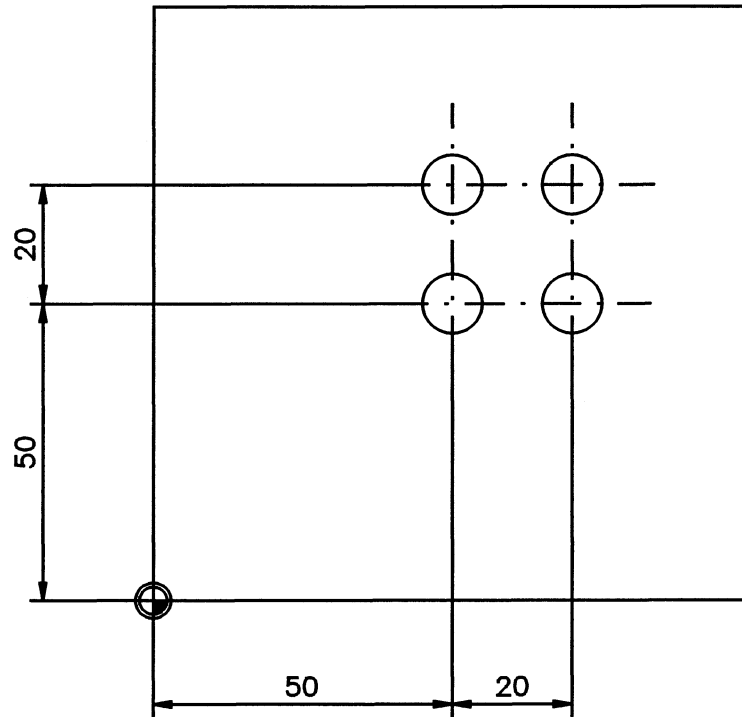
The wordwise absolute and incremental programming does not depend on the modally valid system of measurement 90/G91.

Polar coordinates:

Programming with polar coordinates will not be influence.

Example

EXAMPLE 1.



```

N88550
N1 G17
N2 G54
N3 G195 X0 Y0 Z60 I100 J100 K-80
N4 S1300 T1 M6 (Drill R5)
N5 G0 X0 Y0 Z50
:
N8 G81 Y2 Z-10 F200 M3
N9 G79 X50 Y50 Z0
N11 G79 Y91=20
N12 G79 X91=20
N13 G79 Y91=-20
:
N17 G0 X0 Y0 Z50
N18 M30
    
```

Explanation:

```

N1 : Set the plane to be the XY-plane
N2 : Set the zero point
N3 : Graphic window definition
N4 : activate Tool 1
N5 : Move tool with rapid speed to start position
N8 : Drilling cycle definition
N9 : Drilling the first hole

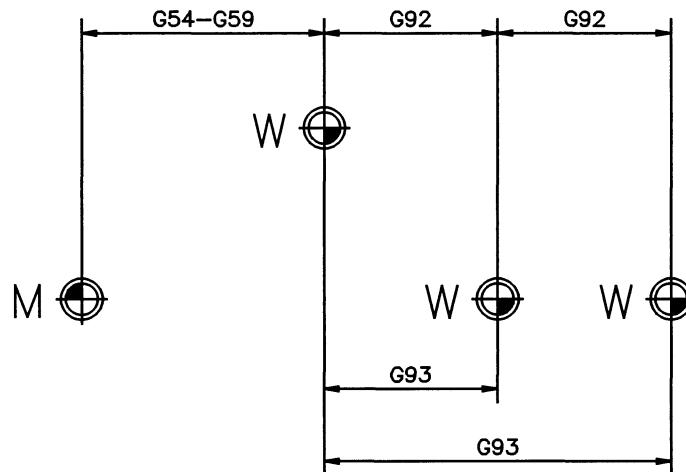
N11 : Incremental Y movement to the second hole
N12 : Incremental X movement to the third hole
N13 : Incremental Y movement to the third hole

N17 : Retract tool
N18 : End of program
    
```

55. Incremental/Absolute zero point shift G92/G93

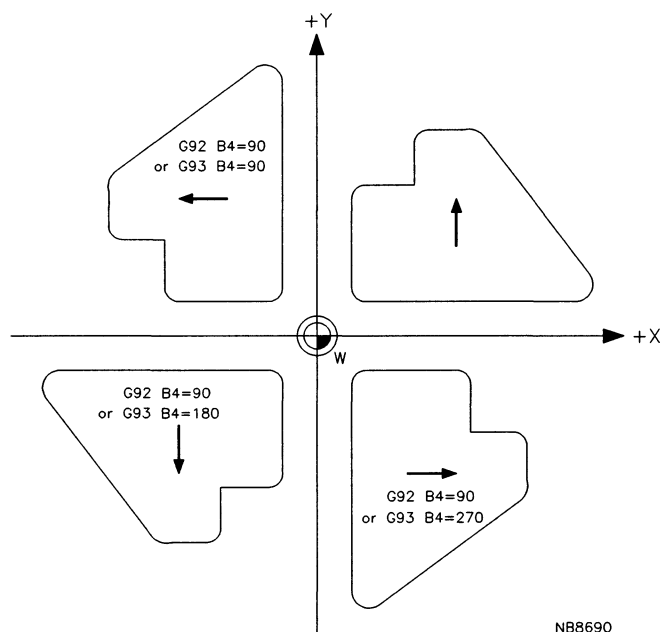
Purpose

1. To establish a program zero point (W) on the workpiece so that workpiece dimensions can be used directly for programming tool or workpiece movements. (G54-G59 or G54I[nr.])



- G92: Incremental zero point shift; shift values are related to last program zero point.
 G93: Absolute zero point shift; shift values are related to a fixed zero point (the mounting zero point C).

2. To produce a rotation of coordinate axes for a group of coordinates



NB8690

- G92: Rotation angle related to the last programmed main axis
 G93: Rotation angle related to a fixed machine tool.

Format

Zero point shift

N... G92 [Coordinate(s) related to last zero point]

N... G93 [Coordinate(s) related to a fixed zero point]

Axis rotation

N... G92/G93 B4=...

Parameters

Words used with G92

1. Zero point shift

X,Y,Z Zero point coordinate

A,B,C Zero point angle

B1= Angle

L1= Path length

2. axis rotation

B4= Angle of rotation incremental

Words used with G93

1. Zero point shift

X,Y,Z Zero point coordinate

A,B,C Zero point angle

B2= Polar angle

L2= Polar length

P,P1= Point definition number

2. Axis rotation

B4= Angle of rotation absolute

Associated functions

With zero point shift: G51/G52, G53-G59 or G54I

With axis rotation: G72/G73

Type of function

Non-modal.

Notes and usage
ZERO POINT SHIFTS**INCREMENTAL ZERO POINT SHIFT (G92)**

The function G92 is used to shift the zero point from the actual program zero point (W) to a new program zero point (W).

APPLICATION OF G92

The function G92 is useful when programming identical tool movements which are repeated at different locations on a workpiece.

ABSOLUTE ZERO POINT SHIFT (G93)

The function G93 is used to establish the program zero point by shifting the zero point from the mounting zero point (C) to the required program zero point (W).

UNALTERED AXIS

When a zero point shift is made and an axis is not involved, that axis does not need to be included in the block where the shift is programmed.

PROGRAMMED DIMENSIONS

All programmed dimensions which follow a zero point shift are measured from the new zero point (0,0).

DISPLAYED COORDINATES

Displayed axis coordinates are always related to the active program zero point W.

CANCELLATION OF G92 AND G93

The G92 zero point shift is cancelled, if a G93 is programmed.

A programmed zero point shift (G92 or G93) is cancelled if another zero point shift function (G51/G52, G53-G59 or G54I[nr.]) is programmed.

A programmed zero point shift (G92 or G93) is cancelled at end of program or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM.

G92/G93 IN TEACH IN MODE

A zero point shift which is input in the TEACH-IN mode is cancelled by CLEAR CONTROL or by the G51 to G59 functions. The G92/G93 function must therefore be included in the part program after the functions G51/G52 or G53-G59.

ROTATION OF AXES

The main plane axes can be rotated around the program zero point W. In this way a part program or a section of a part program can be rotated.

The programmed Coordinates refer to the rotated axes.

PLANE SELECTION

Axis rotation is performed in the active main plane, thus:

G17: X- and Y-axis are rotated
G18: X- and Z-axis are rotated
G19: Y- and Z-axis are rotated

ANGLE OF ROTATION

The angle of rotation is programmed with the word B4=.. The angle ranges from -360° to 360° and is measured as with polar coordinates.

INCREMENTAL ANGLE OF ROTATION (G92)

With G92 the angle is measured with the last active coordinate axis:

- with G17 or G18: the (rotated) X-axis
- with G19: the (rotated) Z-axis

ABSOLUTE ANGLE OF ROTATION (G93)

With G93 the angle is measured with the machine tool axis:

- with G17 or G18: the fixed X-axis
- with G19: the fixed Z-axis

ZERO POINT SHIFT AND AXIS ROTATION

In a G92 or G93 block a zero point shift and axis rotation is allowed. The order of execution is:

- first the zero point shift,
- then axis rotation.

The new zero point is the centre of rotation.

PROGRAMMED ZERO POINT SHIFT AFTER AXIS ROTATION

Once axis rotation is activated, a zero point shift programmed with G92/G93 is not allowed any more.

G51-G59 ZERO POINT SHIFTS

If one of the G-functions G51 to G59 is programmed after axis rotation, the function is executed in the non rotated axes.

MIRROR IMAGE AND SCALING

A combination of mirror image and/or scaling and axis rotation is allowed. The order of execution is:

- first scaling and mirror image,
- then axis rotation.

DISPLAYED COORDINATES

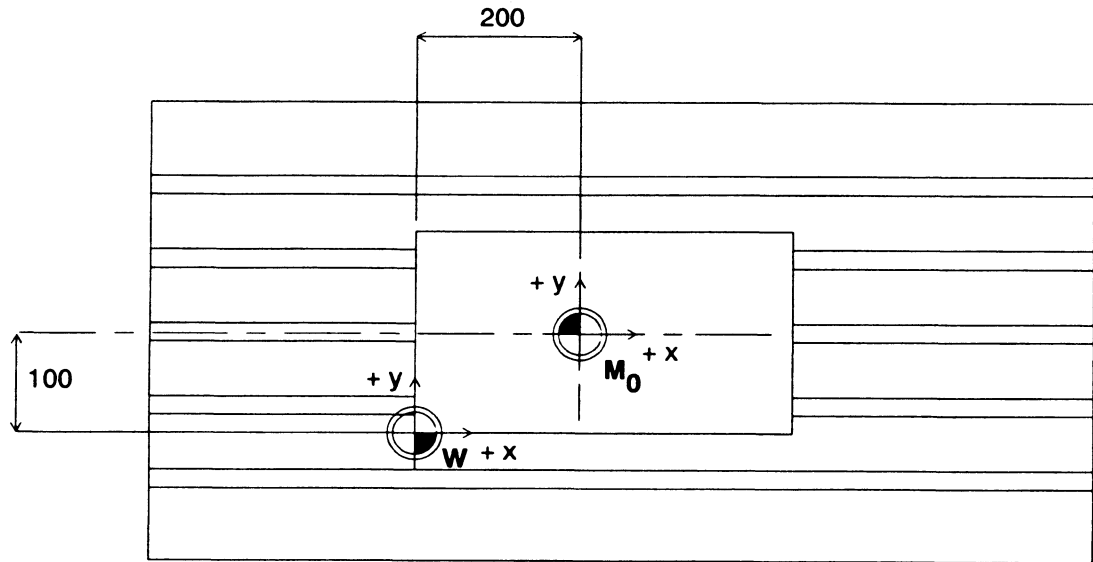
After axis rotation the displayed axis coordinates are related to the non rotated axes of the main plane.

CANCELLATION

The G92 axis rotation is cancelled, if a G93 axis rotation is programmed.

The G93 axis rotation is cancelled with G93 B4=0.

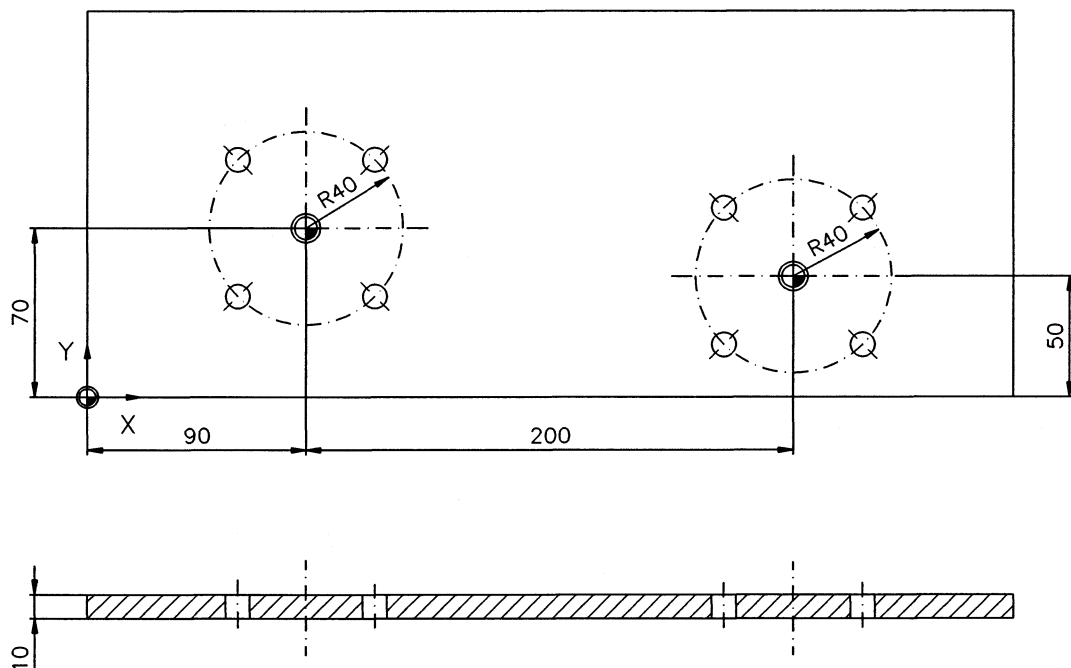
Both axis rotations (G92 and G93) are cancelled at end of program or by softkey CLEAR CONTROL, M30 or by CANCEL PROGRAM.

Examples**EXAMPLE 1.****NB8556**

In this example the centre of the workpiece coincides with the machine zero point (M_0), and the program zero point (W) is to be located in the left hand corner of the workpiece. A G93 can be used to set the program zero point:

```
N30 G93 X-200 Y-100
```

EXAMPLE 2



The four holes around point A and the four holes around B are to be drilled. In the program the program zero point (W) is located at A or at B. In this way calculations during programming are reduced to a minimum.

Program with G92

```

N79560
N1 G17
N2 G54
N3 G195 X-10 Y-10 Z10 I420 J180 K-30
N4 G99 X0 Y0 Z0 I420 J160 K-10
N5 F200 S3000 T1 M6
N6 G92 X90 Y70
N7 G81 Y1 Z-12 M3
N8 G77 X0 Y0 Z0 I45 J4 R40
N9 G92 X200 Y-20
N10 G14 N1=8
N11 G92 X-290 Y-50
N12 G0 Z100 M30

```

Explanation:

N1 : Set the plane to be the XY-plane
 N2 : Set the zero point
 N3 : Graphic window definition
 N4 : Define the blank of the workpiece as a box
 N5 : activate Tool 1
 N6 : Incremental zero point shift
 N7 : Drilling cycle definition
 N8 : Drilling the four holes on the circle
 N9 : Incremental zero point shift
 N10 : Repeat programblock 8. Drilling the four holes on the circle
 N11 : Incremental zero point shift back to the first zero point shift
 N12 : End of program

Program with G93

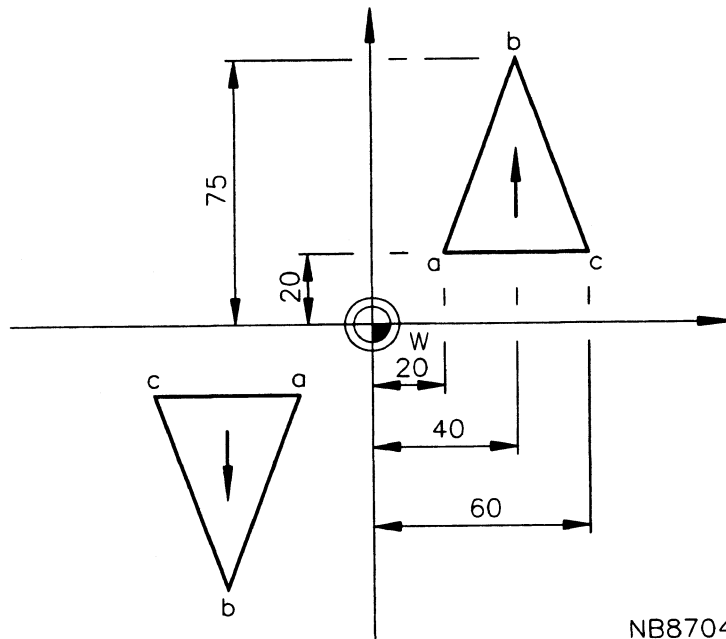
Related to the mounting point C, the program looks as follows:

```
N79561
N1 G17
N2 G54
N3 G195 X-10 Y-10 Z10 I420 J180 K-30
N4 G99 X0 Y0 Z0 I420 J160 K-10
N5 F200 S3000 T1 M6
N6 G93 X90 Y70
N7 G81 Y1 Z-12 M3
N8 G77 X0 Y0 Z0 I45 J4 R40
N9 G93 X290 Y50
N10 G14 N1=8
N11 G93 X0 Y0
N12 G0 Z100 M30
```

Explanation:

```
N1 : Set the plane to be the XY-plane
N2 : Set the zero point
N3 : Graphic window definition
N4 : Define the blank of the workpiece as a box
N5 : activate Tool 1
N6 : Absolute zero point shift
N7 : Drilling cycle definition
N8 : Drilling the four holes on the circle. The surface of the workpiece is define as Z=0.
N9 : Absolute zero point shift
N10 : Repeat programblock 8. Drilling the four holes on the circle
N11 : Absolute zero point shift back to the first zero point shift
N12 : End of program
```

EXAMPLE 3. Rotation of axes



```

N9300
N1 G17
N2 G54
N3 S400 T1 M6
N4 G0 X60 Y-10 Z1 M3
N5 G1 Z-16 F1000
N6 G43 Y20
N7 G41
N8 G1 X20
N9 X40 Y75
N10 X60 Y20
N11 Y-10
N12 G40
N13 G0 Z10
N14 G93 B4=180
N15 G14 N1=4 N2=12
N16 G0 Z100
N17 M30

```

Explanation:

N1:	Set the XY-plane
N2:	Set the stored zero offset
N3:	Load tool 1 and set the spindle speed
N4:	Move tool to start position. Make spindle rotate clockwise at 400 rev/min
N5:	Move tool to depth at set feedrate
N6:	Move tool to point C
N7-N11:	Set radius compensation LEFT and cut workpiece
N12:	Cancel radius compensation
N13:	Move tool rapidly out of workpiece
N14:	Rotate axes through 180 degrees
N15:	Repeat instructions given in block N4 to N12
N16:	Retract the tool from the part
N17:	Cancel rotation of axes and end of program

56. Select feedrate unit G94/G95

Purpose

To control how the CNC interprets programmed (F-word) feedrate values.

G94: Feed rate in mm/min or inches/min.

G95: Feed rate in mm/rev or inches/rev.

Format

N... G94/G95 F...

Parameters

F	Feed
S	Speed (rev/min)
T	Tool number
M	Machine function

Type of function

Modal

Notes and usage

DEFAULT MODE

G94 is automatically made active when the CNC is switched on or by softkey CLEAR CONTROL, M30 or by CANCEL PROGRAM.

DIMENSIONAL UNIT

The dimensional unit for both functions is determined by the functions G71 (metric) or G70 (inches).

CONVERSION TO A FEEDRATE IN UNITS/MIN

When G95 is active the CNC automatically converts the F-value to a feed rate in mm/min (inches/min). If a spindle transducer is fitted to the machine, the measured spindle speed is used for this calculation.

Example

N.. G94

N.. G1 X.. Y.. F200

The tool is moved to a point defined by the coordinates X.. and Y.. at a feed rate of 200 mm/min

N.. G95

N.. G1 X.. Y.. F.5

The tool is moved to a point defined by the coordinates X.. and Y.. at a feed rate of .5 mm/rev.

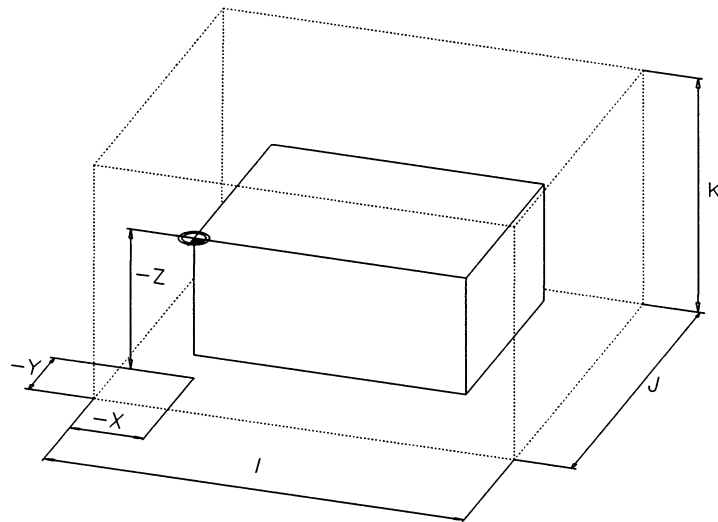
Select feedrate unit G94/G95

57. Graphic window definition G98

It is recommended to start a new program always with G195.

Purpose

To define the position relative to the zero point W and the dimensions of a 3D graphics window in which the machining of a workpiece is to be represented with the graphical simulation of a part program run on the display of the control.



3D graphic window definition

Format

N... G98 X... Y... Z.... J... I... K... {B...} {B1=...} {B2=...}

Parameters

X,Y,Z	Start point coordinate
I	Dimension parallel to X
J	Dimension parallel to Y
K	Dimension parallel to Z
B	Rotation around hor. axis (3D)
B1=	Rotation around vert. axis (3D)
B2=	Rotation around third axis (3D)

Associated functions

G99, G195 to G199

Type of function

Non-modal.

Notes and usage**GRAPHICAL SUPPORT**

Refer to the appendix GRAPHICAL SUPPORT at the end of this manual for a short overview about the graphical support provided in the CNC PILOT control system and to the user manual for using the graphical support.

GRAPHIC WINDOW

The window, thus a bounded area on the display, is a rectangular 3D box which dimensions are defined by the G98-function. The dimensions of the workpiece are defined in a G99 block.

The window is used with the graphical simulation, but also with the synchron graphics with which the actual tool movements on the machine can simultaneously be seen on the display of the control.

DEFAULT WINDOW DIMENSIONS

If the 3D window dimensions are not defined the CNC uses the limit switches distances as default values.

TOOL IMAGE

A tool image can be assigned to the tool with the aid of the G-word in the tool memory. The required image can be selected from a set of available tool images and is used by the CNC system to accurately simulate the machining.

ANGLE OF VIEWING (B, B1=, B2=)

With the simulation graphics or the 3D-wire plot the workpiece can be seen rotated. The angles for viewing the rotated workpiece on the display are defined by the words B, B1= or B2=.

	B	B1=	B2=
	rotation about	rotation about	rotation about
XY-plane (G17)	X-axis	Y-axis	Z-axis
XZ-plane (G18)	Z-axis	X-axis	Y-axis
YZ-plane (G19)	Y-axis	Z-axis	X-axis

Other methods are available for selecting an angle of viewing and are described in the user manual.

DEFAULT SETTINGS FOR ANGLES OF VIEWING

If the angles of viewing are not programmed the following default settings are automatically used by the control:

B60, B1=30, B2=0

Note: The function G98 must be programmed before G99.

It is recommended to start a new program always with G195.

Example

```
N9000
N1 G98 X-20 Y20 Z-25 I140 J-90 K85
N2 G99 X-15 Y15 Z-20 I130 J-80 K75
```

Explanation:

N1: Define the start point and dimensions of the 3D graphic window.

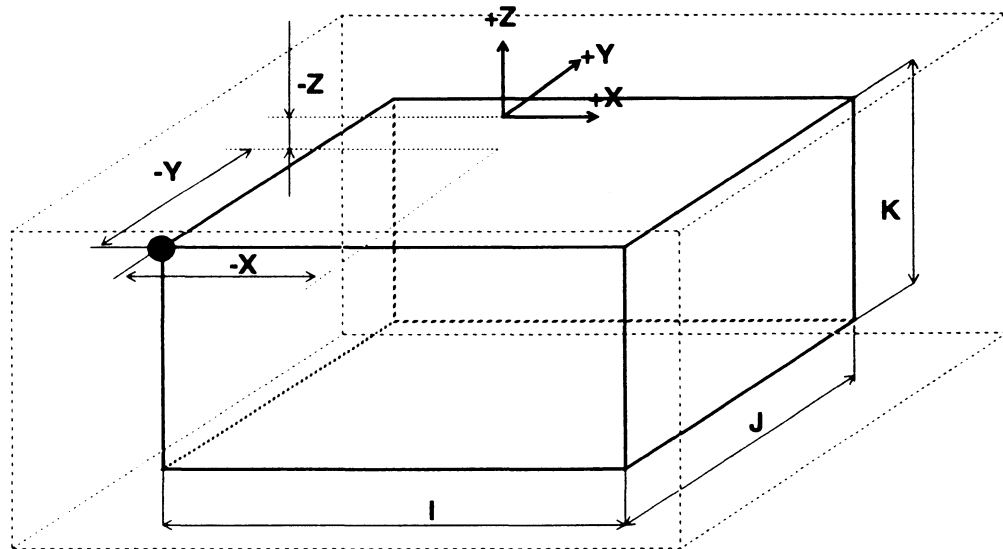
N2: Define the blank of the workpiece as a box.

58. Definition of workpiece blank as a box G99

Purpose

To define the dimensions of a 3D box used as a blank (uncut) workpiece and the position of this 'blank' relative to the program zero point W. These dimensions are used in a graphical simulation of a partprogram run. (This function is used together with the G98-function).

Format



NB8766

N... G99 X... Y... Z... I... J... K...

Parameter

X,Y,Z Start point coordinate
 I Dimension parallel to X
 J Dimension parallel to Y
 K Dimension parallel to Z

Associated functions

G98, G196 to G199

Type of function

Non-modal

Notes and usage**IRREGULAR WORKPIECE SHAPE**

If the blank of the work piece cannot be defined with one box, the functions G196 to G199 must be used to define a workpiece which has an irregular shape.

RESTRICTION

Only one G99 function is allowed in a part program.

Availability of the G99 function enables the part programs developed for the CNC 3000 to be used on the CNC Pilot as well. Unlike the CNC 3000, the program of the CNC Pilot can only contain a single G99 function. If a particular CNC 3000 program includes several G99 functions, the contour should be reprogrammed, using the functions G196 up to G199.

Note: The function G98 must be programmed before G99.

It is recommended to start a new program always with the functions G196 till G199.

Example

```
N9000
N1 G98 X-20 Y-20 Z20 I140 J90 K-85
N2 G99 X0 Y0 Z0 I100 J50 K-55
```

Explanation:

N1: Define the dimensions of 3D graphical window.
N2: Define the blank as a box

59. 3D-Tool correction G141

Purpose

Permits corrections for the toolsize of a 3D-tool path programmed with end point coordinates and normalized vectors in these points and perpendicular to the surface.

Format

To activate 3D-tool correction

N.. G141 {R..} {R1 =..}

To program linear movements

N.. G0/G1 [Coords of end point] [I.. J.. K..]

To cancel 3D-tool correction

N.. G40

Parameters

With G141

R Nominal tool radius

R1= Nominal tool corner radius

The R.. and R1=.. values should be the same as the nominal tool dimensions used by the programming system for calculating the tool path. These values are set equal to zero, if not programmed.

With G0/G1

[Coords of endpoint] [I,J,K]

X,Y,Z,U,V,W Linear coordinates of end point

I,J,K Axes's components of the normalized vector.

Modal words

F, S, some M-functions

General principles

Milling a 3D-surface is executed by moving a specified tool in straight line motions with a specified tolerance along the surface.

Calculating the tool path on a 3D-surface involves a lot of calculations which are usually executed by a NC-programming system or a CAD-system.

The so calculated tool path depends on the tool shape, the tool dimensions and the tolerance on the surface. Running such a part program is only possible with a cutter with the same dimensions as used with the calculations, thus nominal cutters have to be used. If during the machining of the 3D-surface a new tool is needed, this tool should also have the same dimensions as the nominal tool.

If deviations on the dimensions of the work piece are encountered, a new calculation on the programming system must be made.

The 3D-tool correction offers the possibility to use tools which dimensions deviate from the dimensions of the nominal cutter. The corrections are carried out using normalized vectors which are generated together with the end point coordinates by the programming system.

It is also possible to let the programming system calculate the workpiece dimensions and let the control calculate the toolpath from the normalized vectors and the tool dimensions.

Associated Functions

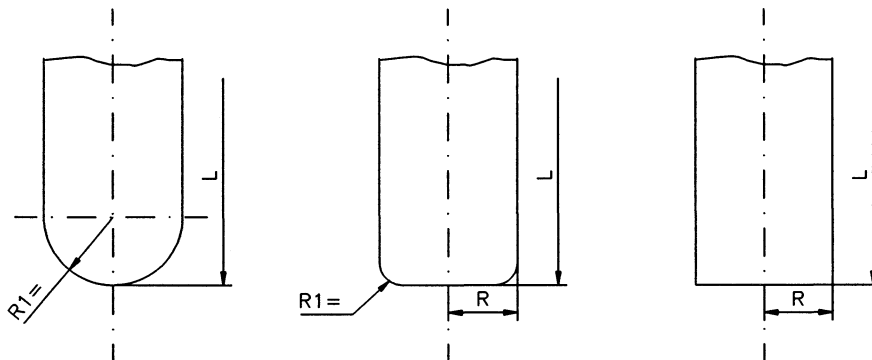
G40 and for radius compensation in a plane G41 - G44

Type of function

Modal

Notes and usage

POSSIBLE TOOLS



Tools used with G141 function

TOOL MEMORY

The following dimensions need to be stored in the tool memory for the use of different types of tools:

Ball cutter: R (=radius of tool), L (=length of tool), C (=radius of tool)

Torus cutter: R (=radius of tool), L (=length of tool), C (=tool corner radius)

Cylinder cutter: R (=radius of tool), L (=length of tool), C0

If a C-value is not stated, it is automatically set equal zero. The default type of cutter is therefore a cylinder cutter.

Note: The corner radius in the G141-block is programmed with the word R1=.
The word C is used for storing the corner radius in the tool memory.

GENERATED TOOL PATH

If the programming system generates the tool path, the dimensions of the nominal tool (R.. and R1=..) are programmed in the G141 block. The tool dimensions stored in the tool memory are used by the control for making a correction on the tool path.

WORKPIECE DIMENSIONS

If the programming system generates the workpiece dimensions, the words R.. and R1=.. are not programmed in the G141 block. The tool dimensions stored in the tool memory are used by the control for calculating the tool path.

ACTIVATING G141

In the first block after the G141 the cutter moves from the actual tool position to the corrected position of that block.

COORDINATES

Only absolute or incremental cartesian dimensions can be used. The coordinates in the first G141 block must be absolute and measured from the program datum point W.

If the parallel linear axes U, V and W are available, they can be used in stead of X, Y and Z.

G90/G91

The functions G90 and G91 are used for programming absolute (G90) or incremental (G91) dimensions. These functions must be alone in their own block.

MIRROR IMAGE

If mirror image (G73 and axis coordinate) is active before the function G141 is activated, the mirrored coordinates are used during the 3D-tool correction.

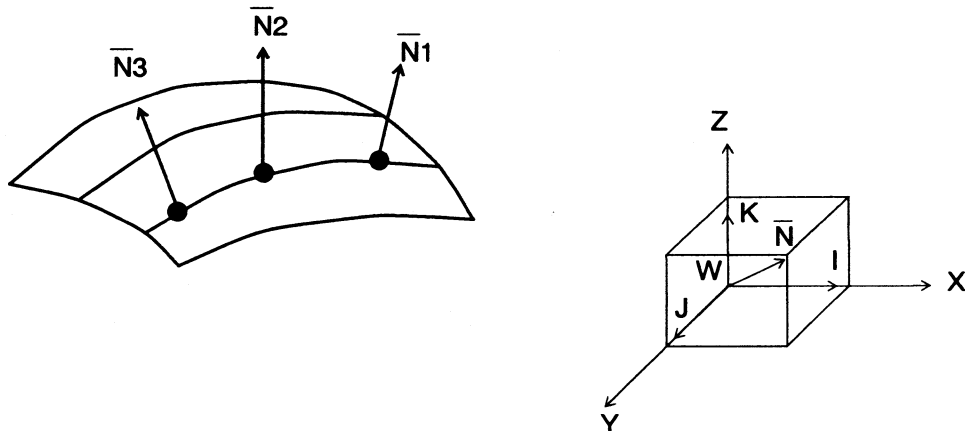
Once G141 is activated, mirror image is still possible. Cancelling mirror image must be done via the G73.

RADIUS COMPENSATION G41 - G44

Active radius compensation programmed with G41 to G44 is cancelled when a G141 block is activated.

CALCULATION METHOD

A normalized vector (= a vector perpendicular to the surface) should be calculated for each end point. A normalized vector is always equal to 1. The I, J and K words are used to program the vector components in the X, Y and Z-axis.



NB8695a

The axes vector components are independent of the selected plane.

If a vector component is omitted in a block, the last programmed value is used.

If a component is not programmed in the first block with a movement after the G141-block, the non-programmed component is set zero.

SCALING FACTOR

The input format of the I, J and K-word is limited to three decimal positions after the decimal point. If this does not produce sufficiently accurate calculations, the I, J and K - values can all be multiplied by a scaling factor to increase dimensional accuracy. The scaling factor is a value between 1 and 1000. So with a factor of 1000 the input accuracy of the vector components is increased to six decimals. This multiplication of the vector components must be performed in the NC programming system.

The control automatically makes the necessary alterations to produce the correct vector components.

Blocks with different scaling factors can be combined in the same part program. The function G141 must be programmed before each change of the scaling factor.

UNDERCUTS

Undercuts or collisions between tool and material at non-cutting points are not detected by the control.

CANCELLATION

The G141 function is cancelled by either the function G40 or by softkey CLEAR CONTROL, M30 or by softkey CANCEL PROGRAM. The cutter halts at the last corrected position.

FUNCTIONS TO BE CANCELLED

The functions G64, scaling (G73 A4=..), axes rotation (G92/G93 B4=..) and G182 should be cancelled, when G141 is programmed.

FUNCTIONS PERMITTED

G0, G1, G4, G22, G29, G40, G73, G90, G91, G141

G0: Positioning logic not executed.

G4: Dwell time programmed with X-word. No other words allowed.

G22: Macro call to be used with BTR

G29: Condition for the jump. E-parameter must be set before the G141-block or in the G29-block.

G73: mirror image only

PROGRAMMING RESTRICTIONS

All G-functions not mentioned, are not permitted.

The use of defined points (P) and E-parameters is not permitted too.

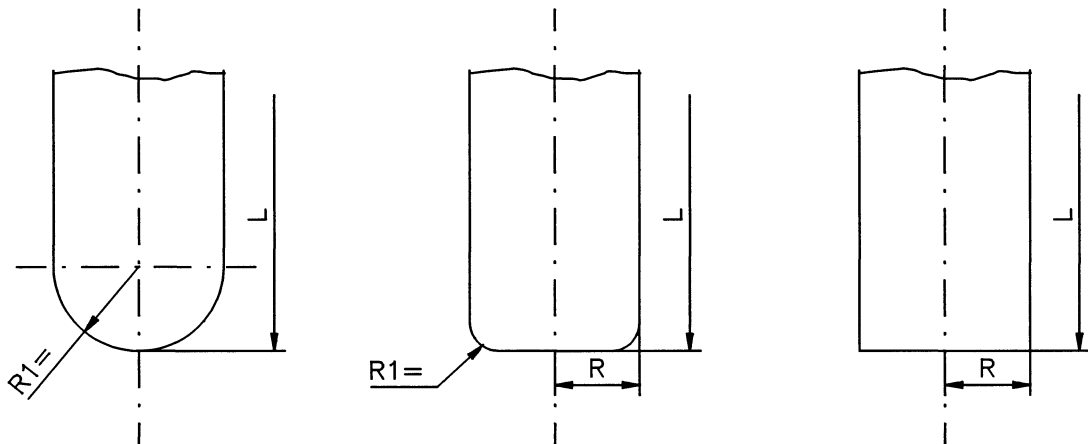
A tool change is not allowed, once G141 is activated.

OPERATING RESTRICTIONS

G141 is not allowed with TEACH IN and Workpiece Presentation.

Note: It is advised not to use 3D-tool correction with Erase Graphics.

The formulae given below can be used for calculating the required dimensions when the G141-function is to be used for 3-D tool correction



Ball cutter ($R1=R$)

$$X_c = X_p + R \cdot I$$

$$Y_c = Y_p + R \cdot J$$

$$Z_c = Z_p + R \cdot (K-1)$$

Torus Cutter ($R > R1$)

$$X_c = X_p + C \cdot I + (R-C) \cdot I / \sqrt{I^2 + J^2}$$

$$Y_c = Y_p + C \cdot J + (R-C) \cdot J / \sqrt{I^2 + J^2}$$

$$Z_c = Z_p + C \cdot (K-1)$$

Cylinder Cutter (R1=0)

$$\begin{aligned}X_c &= X_p + R \cdot I / \text{SQRT}(I^2 + J^2) \\Y_c &= Y_p + R \cdot J / \text{SQRT}(I^2 + J^2) \\Z_c &= Z_p\end{aligned}$$

R - Cutter radius

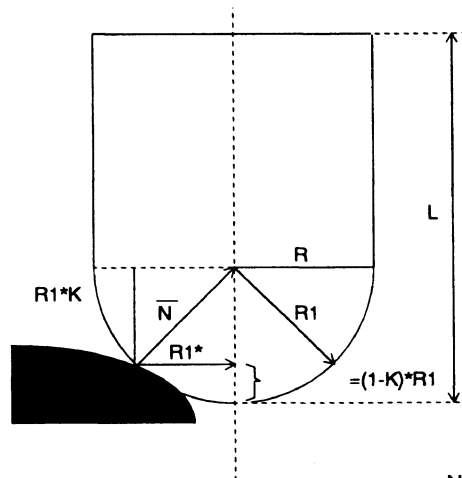
R1 - Rounding radius

Xc,Yc,Zc - Corrected end points

Xp,Yp,Zp - Programmed end points

I,J,K - Axes components of normalized vector **N**

The above formulae apply when the XY-plane is active. If the XZ or ZY-plane is selected the CNC will make the necessary alterations. The axes values of the normalised vector are independent of the selected plane.

Example

NB8695a

```

...
N19
N20 G141 R.. R1=.. F..
N21 G1 X.. Y.. Z.. I.. J.. K..    (first measure vector)
.
.
.
N300 G141 R.. R1=.. F..
N301 G1 X.. Y.. Z.. I.. J.. K.. (second measure vector)
.
.
.
N2400 G141 R.. R1=.. F..
N2401 G1 X.. Y.. Z.. I.. J.. K.. (third measure vector)
.
.

```

Explanation:

R.. R1=.. : The used radius to calculate the position and the vector in a CAD-system. During milling the actual radius in the Tool-table is used.

X.. Y.. Z.. : End point coordinates

I.. J.. K.. : Axes's components of the normalized vector multiplied by a measure vector.

60. Linear measuring movement G145

Purpose

To execute a free programmable linear measuring movement to determine axis positions in measuring cycle macros. Measuring a position in one axis only can also be executed with the function G45.

Format

N... G145 [point to be measured] [(axis address) 7=...] E... {F2=...} {K... } {L...}

Parameters

X,Y,Z Endpoint coordinate
 A,B,C Endpoint angle
 B1= Angle
 B2= Polar angle
 L1= Path length
 L2= Polar length
 P,P1= Point definition number

(axis address) 7=

X7= E-par. for measured value in X
 Y7= E-par. for measured value in Y
 Z7= E-par. for measured value in Z
 A7= E-par. for measured value in A
 B7= E-par. for measured value in B
 C7= E-par. for measured value in C

Measuring

K K0:tool correction on / K1:off
 L 0:measure by contact,1:by release
 E E-parameter for measure status
 F2= Measure feed

For absolute and incremental programming

X90=,Y90=,Z90= Absolute endpoint
 A90=,B90=,C90= Absolute endpoint angle
 X91=,Y91=,Z91= Incremental endpoint
 A91=,B91=,C91= Incremental endpoint angle

Associated Functions

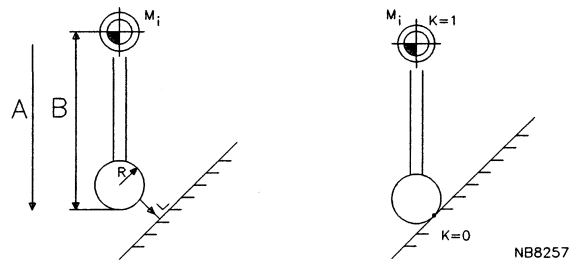
G148, G149, G150 and the measuring cycles G45 and G46 M24, M26, M27, M28, M29
 Wordwise absolute/incremental programming (X90=..., X91=..)

Type of function

Non modal

Notes and usage

COMPENSATING FOR PROBE DIMENSIONS (K)



A: Tool axis
B: Tool length

- K0: Tool correction on. Measured positions are corrected for the tool length and radius. Measured positions of rotary axes are not corrected for tool data.
K1: Tool correction off. Measured positions are not corrected. If K is not programmed, K0 is automatically activated.

The following assumptions are used when the measured positions are corrected for the probe dimensions.

- The probe is parallel to the tool axis.
- The probe is perfectly spherical.
- The probe movement is perpendicular to the surface being measured.

MEASURING CONDITIONS (L)

- L0: Measure at contact.
L1: Measure at probe release from the measuring surface (see example).
If L is not programmed, L0 is automatically used.

STORAGE MEASURED VALUE [(Axis address)7=]

This word states the E-parameter number which is to contain the measured axis position; eg. X7=2, states that X-axis measured value must be stored in parameter E2; X7=E1 (E1=5) means measured value is stored in E5.

MEASURING PROBES STATUS (E)

The measuring probe can be in one of three states after completion of a G145 block. The assigned E-parameter can therefore have one of three states.

- E... = 0: the programmed position has been reached, but no measuring point has been found. The assigned E-parameters, which contain measuring values, remain unaltered.
- E... = 1: during the measuring movement a measuring point has been found. The measured position of the axes has been entered into the E-parameters.
- E... = 2: the G145 block was executed while Block Search, Testrun or the Demo mode was active.

MEASURING FEEDRATE (F2=)

If F2= is not programmed a default value stored in a machine constant(MC843) is used automatically.

Note: If a function for spindle direction (M03 or M04) is entered, this function will be suppressed and an error code output.

The function G145 is not permitted, when G182 is active.

BLOCK SEARCH

During BLOCK SEARCH the measuring movement is simulated. The E-parameters in which the measured coordinates were to be loaded, remain unaltered. The signals from the measuring probe are ignored.

DEMO

During DEMO a movement is executed towards the programmed position. The programmed coordinates are loaded into the E-parameters. The signals from the measuring probe are ignored.

TESTRUN

During TESTRUN the measuring movement is executed with the test feed rate (a Machine Constant (MC741)) or simulated if TESTRUN is executed without movements. The programmed coordinates are loaded into the E-parameters. When the probe is triggered during a movement, the movement is aborted and a collision error generated.

GRAPHICS

In the GRAPHICS modes the measuring movements are simulated. The programmed coordinates are loaded into the E-parameters. The signals from the measuring probe are ignored.

Note: In all the mentioned operation modes, the E-parameter for the measuring probe status gets the value 2. By checking this parameter in the measuring macros it is possible to avoid using parameters which do not contain measured values.

INTERVENTION

With INTERVENTION the G145-movement is treated as a G1-movement. The status of the probe should not be changed between the start point of the measuring movement and the point of interrupt. If the status was changed, an error message is displayed. An error message is also displayed, if the probe is triggered during the repositioning.

Examples

EXAMPLE 1:

N90 G0 X25 Y18

N95 M27

N100 G145 X35 Y28 E12 F2=100 X7=25 Y7=26

N105 M28

Explanation:

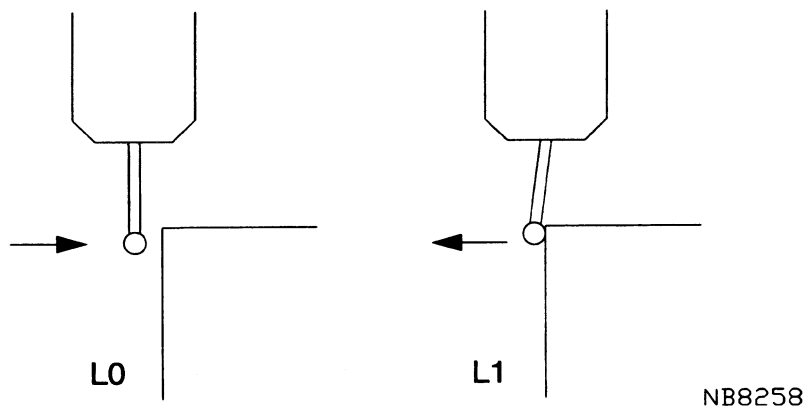
N90: Start position linear measuring movement

N95: Activate measure probe

N100: Measure the point specified by X35, Y28 and store the result of the X-axis and Y-axis in E-parameter numbers 25 and 26 respectively. E-parameter number 12 stores the number indicating the probe status. F2=100 sets the measuring feedrate to 100 mm/ min.

N105: Deactivate measure probe

Remark on programming with L1.



Two successive G145 blocks are to be programmed.

N21 G0 X35

N22 G28 I3=1

N23 G145 L0 X55 F2=2000

N24 G29 E0 E0=E7<1 N=26

N25 G145 L1 X45 X7=13 F2=500

N26

Explanations:

N21: Start position linear measuring movement

N22: It is advisable to program the positioning function, in order that the switching point position be definitely reached.

N23: Movement at high feed rate to the measuring surface. L0 must be programmed.

N24: Check to see if the box is reached

N25: The probe measures the programmed point when moving away from the surface. The measured X-coordinate is stored in parameter I3.

EXAMPLE 2: Measuring tool length with a measuring box

Two macros and a program are given for measuring the tool length with the aid of a measuring box. In the first macro (N14501) the trigger point of the box is determined. In the second macro (N14502) the actual length measurement is executed. In the program (N14503) parameters are set and both macros called.

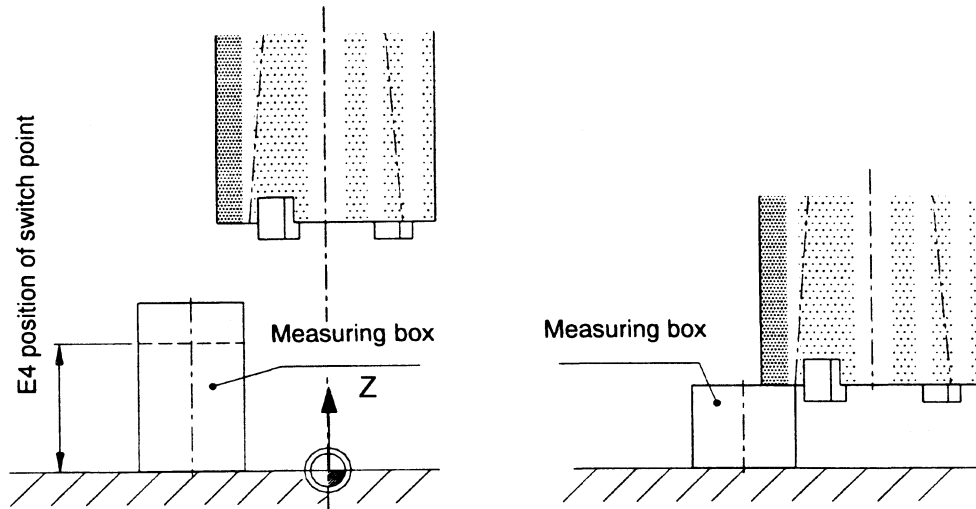
Note

1. Refer to the machine tool builder's documentation to see if a measuring box can be used and which M-functions have to be used for switching on and off the measuring box in a particular installation.
2. The program and macros are given to show the possibilities of the measuring cycles and the E-parameters. Further update might be necessary to adapt the macros to the specific requirements of the user.

Used parameters

E0:	E-Parameter for jump function
E1:	X-coordinate of the measuring box
E2:	Y-coordinate of the spindle reference point
E3:	Z-coordinate of the measuring box
E4:	Y-coordinate of the trigger point of the box
E5	=0: Trigger point is not determined =1: Trigger point is already determined
E7	=0: No error found in macro N14501 =1: An error found in macro N14501
E8:	Tool number or Tool identification number
E10:	Measured Y-coordinate

Macro for determination the trigger point of the measuring box.

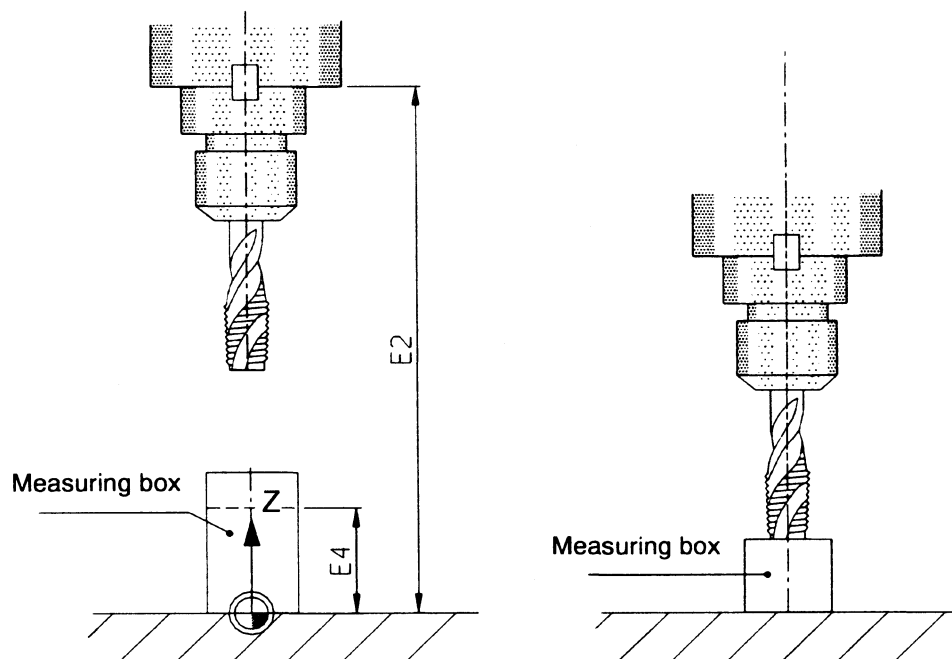


```

N14501
N1 T0 M6
N2 M24
N3 G0 X0 Y0 Z150 E7=0
N4 G28 I3=1
N5 G145 Y20 L0 E7 F2=2000
N6 G29 E0=E7<>1 E0 N=10
N7 G145 Z150 L1 E7 F2=200 Z7=4
NS G0 Z150
N9 G29 E0 E0=1 N=12
N10 M0 (NO TRIGGER POINT FOUND)    E7=1
N11
    
```

Explanation of macro N14501

- N1 : Unload the spindle
- N2 : Activate the measuring box. M24 is an assumed function for activating the box.
- N3 : Move the spindle nose above the measuring box
- N4 : Feed movements should reach their position accurately.
- N5 : Measuring movement in the Z-axis (=tool axis). A large Z-position is used so that it is sure that the box is reached. Measure at contact (L0).
- N6: Check to see if the box is reached. (E7=1!)
- N7: Measuring movement. Measure at release. At that moment the trigger point is found.
- N8: Retract the spindle
- N9: Jump to the end of the macro
- N10: Program stop and an message that the trigger point is not found.

Macro for measuring the tool length.

```

N14502
N1 T=E8 M6
N2 G0 X=E1 Y=E2 Z=E3
N3 G145 Y=(E4-20) L0 E7 F2=2000
N4 G29 E0=E7 <>1 E0 N=10
N5 G145 Y=E2 L1 E7 Z7=10 F2=200
N6 G29 E0=E7 <>1 E0 N=10
N7 G0 Z=E3
N8 G150 T=E8 L1=E10-E4
N9 G29 E0 E0=1 N=12
N10 M0 (LENGTH MEASUREMENT UNSUCCESSFUL)
N11

```

Explanation of macro N14502

N1 : Load the tool to be measured
 N2 : Move the tool above the measuring box
 N3 : Measuring movement in the Z-axis (=tool axis). A Z-position past the trigger point is used so that it is sure that the box is reached. Measure at contact (L0).
 N4: Check to see if the box is reached. (E7=1!)
 N5: Measuring movement. Measure at release. At that moment the length is found.
 N6: Check to see if a measurement took place (E7=1!)
 N7: Retract the tool
 N8: Update length of the actual tool in the tool memory
 N9: Jump to the end of the macro
 N10: Program stop and an message that no length is measured.

Program for measuring tool length with a measuring box

The first time the program is used, the user has:

- to move the spindle nose above the measuring box
- to set the zero point with PRESET AXES
- to enter the parameters E5=0 and E8

If more tools have to be measured, parameter E5=1 has to be set and parameter E8 (the tool number) to be entered.

```

N14503 E8=.. (TOOL NUMBER)
N1 E5=0 (=0 DETERMINE TRIGGER POINT =1 NO TRIGGER POINT)
N2 G17
N3 G54
N4 G29 E5 K0 N=8
N5 G22 N=14501
N6 G29 E0 E0=E7<1 N=9
N7 E1=0 E2=0 E3=350
N8 G22 N=14502
N9 M28
N10 T0 M6
N11 G53
N12 M30
    
```

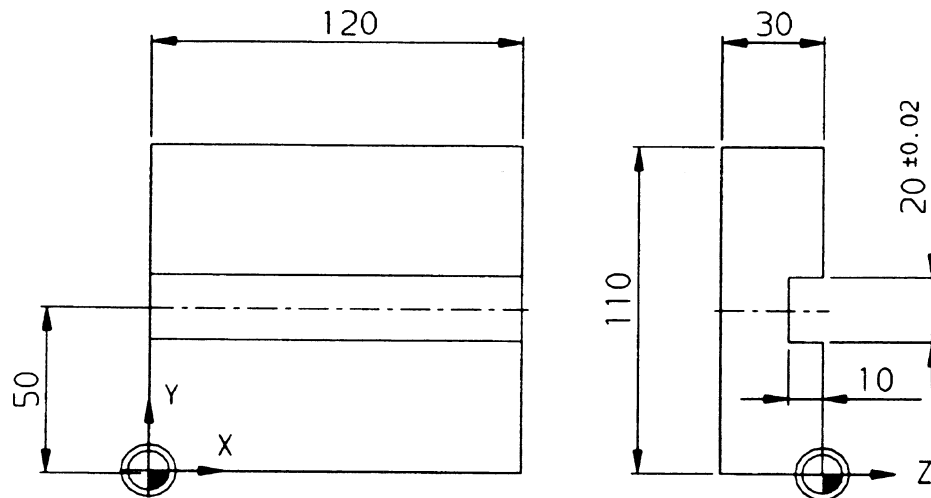
Explanation of the program

```

N14503/N1  Set the parameters E5 and E8
N2:        Set the plane of operation to be the XY-plane
N3:        Set the zero point found with PRESET AXES
N4:        Check to see if the trigger point has to be determined (E5=0) or can be ignored.
N5:        Call macro N=14501 to determine the trigger point of the measuring box
N6:        Check to see if an error was found in macro N14501 (E7=1); in that case ignore the next
            macro call.
N7:        Set the parameters containing the coordinates of the box
N8:        Call macro N14502 to measure a tool length
N9:        Switch off the measuring box. M28 is assumed to be used for that purpose.
N10:       Unload the spindle
N11:       Restore the zero point
N12:       End of the program
    
```

EXAMPLE 3: Milling and measuring a groove

Milling a groove followed by measuring the width of the groove. If the width is too small, the radius of the mill is changed and the groove is finished again.



N14504 (MILLING AND MEASURING A GROOVE)

N1 G17

N2 G54

N3 E15=20.02 (MAXIMUM WIDTH OF THE GROOVE)

N4 E16=19.98 (MINIMUM WIDTH)

N5 E3=(E15+E16):2

N6 S1000 T1 M6 (MILL Diameter 18 mm)

N7 G0 X-40 Y50 Z-10 B0 F400 M3

N8 G1 X140

N9 G43 Y60

N10 G41

N11 X-25

N12 Y40

N13 X140

N14 G40

N15 Y50

N16 G0 Z50 M5

N17 G149 T0 E30

N18 T30 M6 (TOUCH TRIGGER PROBE)

N19 D207 M19

N20 M27

N21 X60 Y50 Z-8

N22 M29

N23 G145 Y65 E10 Y7=1 F2=500

N24 G0 Y50

N25 G29 E11=E10=0 E11 N=29

N26 M29

N27 G145 Y35 E10 Y7=2 F2=500

N28 G0 Y50

N29 M28

N30 G29 E11=E10=0 E11 N=41

N31 E5=E1-E2

N32 E6=(E5-E3):2

N33 G29 E20=E5>E15 E20 N=43

N34 G29 E20=E5>E16 E20 N=45

```

N35 G149 T=E30 R1=4
N36 G150 T=E30 R1=E4+E6
N37 S1000 T1 M6 (MILL Diameter 18 mm)
N38 G0 X140 Y50 Z-10 B0 F400 M3
N39 G29 E20 E20=1 N=43
N40 M0(PROBE NOT TRIGGERED, NO MEASUREMENT EXECUTED)
N41 G29 E20 E20=1 N=43
N42 M0 (GROOVE WIDTH TOO BIG)
N43 M30

```

Explanation

N1: Define the plane of operation
 N2: Activate a stored zero offset
 N3: Set the maximum width of the groove
 N4: Set the minimum width of the groove
 N5: Calculate the width of the groove with the average tolerance
 N6: The mill of 18 mm diameter is loaded as tool 1
 N7: Start the spindle and move tool to start position
 N8: Mill through the centre of the groove
 N9-N13: Mill the sides of the groove
 N14: Cancel radius compensation
 N15: Move the tool to the middle of the groove
 N16: Retract the tool and stop the spindle
 N17: Pick up the number of the actual tool
 N18: Load the touch trigger probe
 N19: Set the probe in an oriented position. It depends on the machine tool if this setting is necessary.
 N20: Activate the probe
 N21: Move the probe to the middle of the groove and at depth
 N22: Activate the air pressure. The code of the M-function depends on the machine tool
 N23: Measure the upper side of the groove.
 Store the measured value in the Y-axis at E1
 Store the status of the measuring probe at E10
 N24: Move the tool back to the middle of the groove
 N25: Check to see if a measurement was executed
 E10=0 no measurement. Jump to N29 to switch off the probe and to display an error message.
 N26: Activate the air pressure.
 N27: Measure the lower side of the groove.
 Store the measured value in the Y-axis at E2
 Store the status of the measuring probe at E10
 N28: Move the tool back to the middle of the groove
 N29: Switch off the probe
 N30: Check to see if a measurement was executed
 E10=0 No measurement. Jump to N41 to display an error message.
 N31: Calculate the actual width of the groove from the measured Y-positions
 N32: Calculate the difference between the programmed width and the measured width. The difference is related to the tool radius.
 N33: Check to see if the measured tool width (E5) is greater than the maximum allowed width (E15). If greater jump to N43 to display an error.
 N34: Maximum width is not exceeded. Check to see if the measured width (E5) is greater than the minimum allowed width. If greater the groove is finished. So jump to the end of the program (N44).
 N35: The measured width is less than the minimum value. In this case the tool radius is changed in the tool memory and the sides of the groove milled again with the new radius value from the tool memory. Read the tool radius from the tool memory and store its value in parameter E4.

N36: Store the recalculated tool radius in the tool memory.
N37: The mill is loaded again
N38: Start the spindle and move tool to start position
N39: If the milling is finished, jump to the end of the program.
N40: Program stop for displaying an error text
N41: If the program continues, a jump to the end of the program is executed
N42: Program stop for displaying an error text
N43: End of program

EXAMPLE 4: Alignment of workpiece mounted on a rotary table

Only two points in X or Y have to be measured to be able to adjust a workpiece mounted on the rotary table rotating around the Z axis.

The angle between workpiece and X axis is calculated automatically by the CNC and may be used to rotate the table for positioning the workpiece parallel to the X axis.

If the workpiece is inclined towards the X axis at the beginning, this angle may be programmed with the C word.

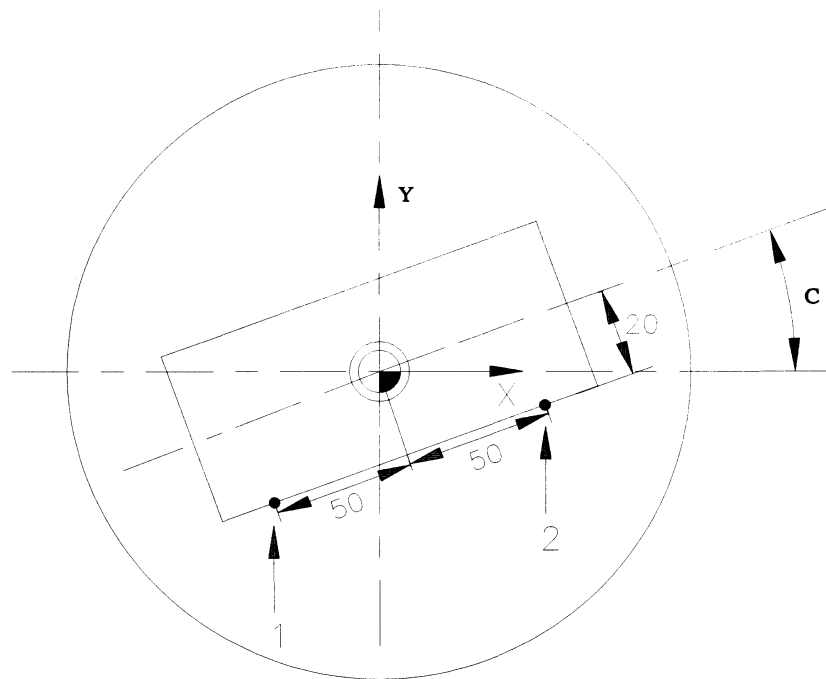
If C has not been programmed, C0 is used automatically instead of C.

Note:

This function can only be used when:

1. the rotary table is located in the XY plane, the workpiece is rotated around the Z axis (C axis) and the measuring probe is in Z direction.
2. the measurements are carried out in Y direction.

A part is mounted on a rotary table and should be aligned parallel to the X-axis. With a touch trigger probe two points on the part are measured and then the table is rotated over the calculated angle.



NB9804

```

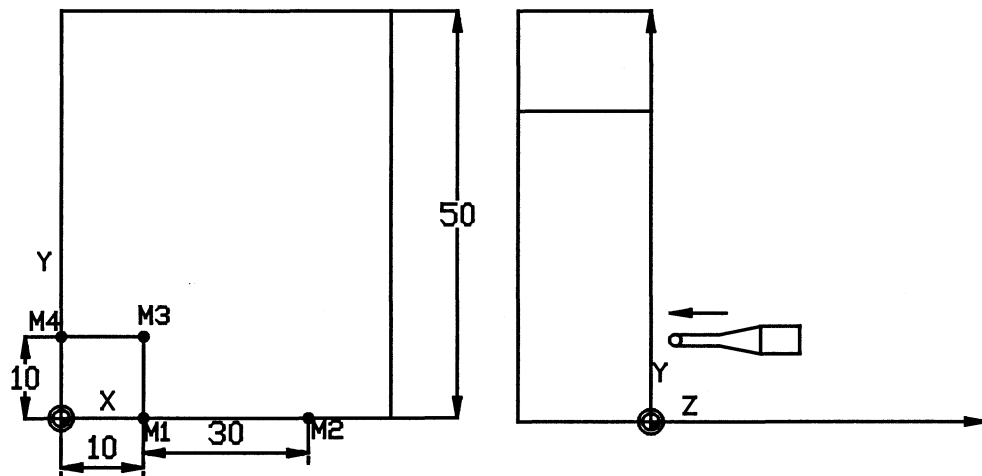
N50003
N1 G17
N2 G54
N3 T1 M6
N4 G0 X-50 Y-30 Z100 C0
N5 G1 Z0
N6 M27
N7 G145 X-50 Y-20 X7=11 Y7=12
N8 G0 Y-30
N9 G0 X50
    
```



```
N10 G145 X50 Y-20 X7=21 Y7=22
N11 G0 Y-30
N12 M28
N13 G0 Z100
N14 E30=(E12-E22)
N15 E31=(E21-E11)
N16 E32=(E30:E31)
N17 E33=atan(E32)
N18 G150 C7=E33 N1=54
N19 G54
N20 G0 C0
N21 M30
```

Explanation:

```
N1 : Set the plane of operation
N2 : Set the zero point
N3 : Load the touch trigger probe
N4 : Positioning first measuring point
N5 : Going to depth
N6 : Activate probe
N7 : Measure point 1 in direction Y (X in E11, Y in E12)
N8 : Retract
N9 : Positioning second measuring point
N10 : Measure point 2 in direction Y (X in E21, Y in E22)
N11 : Retract
N12 : Deactivate probe
N13 : Retract Tool
N14 : Calculate the difference (E30) in Y direction
N15 : Calculate the difference (E31) in X direction
N16 : Calculate Quotient (E32)
N17 : Calculate ARCTAN (E33)
N18 : A angle correction is made in the zero point of the C-Axis
N19 : Set the zero point
N20 : Rotate table back to C0
N21 : Program end
```

EXAMPLE 5 : Determining the zero point


The probe is standing in the Z-axis. The part is mounted on a table rotating around the Z-axis. Five points of the part (M1 to M5) are measured. M1 and M2 for covering the angular displacement; M3, M4 and M5 for measuring the positions of the axes.

The section of the part program for determination the zero point could be:

```

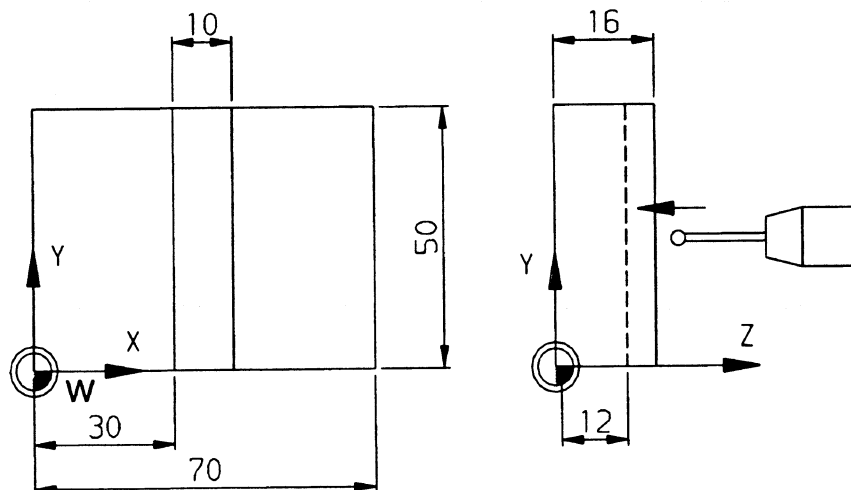
N50004
N1 G54
N2 G17
N3 T1 M6 (probe)
N4 G0 X10 Y-10 Z70 C0 F1000
N5 G1 Z-5
N6 M27
N7 G145 X10 Y0 X7=11 Y7=12
N8 G0 Y-10
N9 G0 X50
N10 G145 X50 Y0 X7=21 Y7=22
N11 G0 Y-10
N12 G0 Z70
N13 E30=(E12-E22)
N14 E31=(E21-E11)
N15 E32=(E30:E31)
N16 E33=atan(E32)
N17 G150 C7=E33 N1=54
N18 G54
N19 G0 C0
N20 G0 X10 Y10 Z10
N21 G145 X10 Y10 Z0 Z7=3
N22 G0 Z10
N23 G0 X-10 Y10
N24 G1 Z-5
N25 G145 X0 Y10 Z-5 X7=1
N26 G0 X-10
N27 G0 Z10
N28 G0 X10 Y-10
N29 G1 Z-5
N30 G145 X10 Y0 Z-5 Y7=2
N31 G0 Y-10
N32 G0 Z50
    
```

```
N33 G150 X7=E1 Y7=E2 Z7=E3 N1=54
N34 G54
N35 M28
:
```

Explanation:

```
N1 : Set the zero point
N2 : Set the plane of operation to be XZ-plane
N3 : Load the touch trigger probe
N4 : Move to programmed position
N5 : move to depth in hole 1
N6 : Activate probe
N7 : Measure point M1 (X in E11, Y in E12)
N8 : Retract probe to avoid collision
N9 : Move to position for measure point 2
N10 : Measure point M2 (X in E21, Y in E22)
N11 : Retract probe to avoid collision
N12 : Retract probe to avoid collision
N13 : Calculate the difference (E30) in Y direction
N14 : Calculate the difference (E31) in X direction
N15 : Calculate quotient (E32)
N16 : Calculate ARCTAN (E33)
N17 : A angle correction is made in the zero point of the C-Axis
N18 : Set the zero point
N19 : Rotate table back to C0
N20 : Move to measure point 3
N21 : Measure point M3, to determine the position in the Tool axis(Z in E3)
N22 : Retract probe to avoid collision
N23 : Move to measure point 4
N24 : Move to depth
N25 : Measure point M4, to determine the position in the X axis(X in E3)
N26 : Retract probe to avoid collision
N27 : Move to measure point 1
N28 : Move to depth
N29 : Measure point M1, to determine the position in the Y axis(Y in E3)
N30 : Retract probe to avoid collision
N31 : Update of the zero offset values in the X-, Y- and Z-Axis
N32 : Set the updated zero point
N33 : Deactivate probe
```

EXAMPLE 5: Correcting the length of a tool



With a mill a groove is made, the depth of the groove is measured and the tool length of the mill updated.

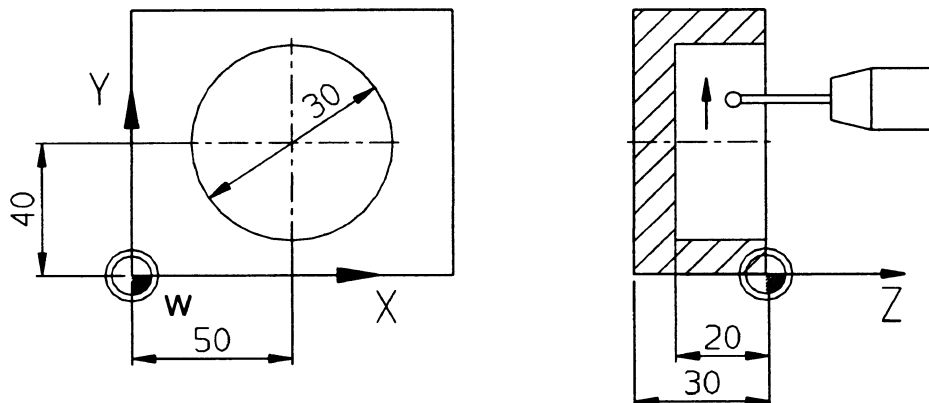
```

N90005
N1 G17
N2 T1 M6 (Mill R5)
N3 G0 X35 Y60 Z12 S1000 M3
N4 G1 Y-10 F200
N5 G0 Z200 M5
N6 T2 M6 (Prober)
N7 G0 X35 Y25 Z20
N8 M27
N9 G145 X35 Y25 Z12 Z7=1
N10 G149 T1 L1=2
N11 G150 T1 L1=E2-E1+12
N12 M28
N13 Z200 M30
    
```

Explanation:

N1 : Set the plane of operation to be the XY-plane
 N2 : Load the mill of 10 mm diameter
 N3 : Start the spindle and move the mill to the start point
 N4 : Mill the groove
 N5 : Retract the tool and stop the spindle
 N6 : Load the probe
 N7 : Move to start point
 N8 : Activate probe
 N9 : Measure the tool axis in negative direction (Z in E1)
 N10 : Store the tool length T1 in E2
 N11 : The tool length is dependent of the calculated difference in the Z-Axis.
 N12 : Deactivate probe
 N13 : Retract probe and end program.

EXAMPLE 6 : Milling and measuring a hole



A hole is milled and measured with a touch trigger probe. Checks are provided to see if the tolerance on the radius of the hole is within the required limits. If the radius is too small, the hole is milled again. If the radius is too large the part is rejected and an message displayed.

The part program could be:

```

N50006
N1 G54
N2 G17
N3 T1 M6 (mill R=5)
N4 E1=50 (X drilling centre point)
N5 E2=40 (Y drilling centre point)
N6 E3=0 (Z drilling start point)
N7 E4=20 (hole depth)
N8 E5=15 (drill radius)
N9 E6=3 (probe radius)
N10 (E7 Measure status)

N11 G89 Z=-E4 B2 R=E5 K6 F300 S1000 M3
N12 G79 X=E1 Y=E2 Z=E3
N13 G0 Z50 M5

N14 T2 M6 (Probe)
N15 M27
N16 G0 X=E1+E5-E6-2 Y=E2 Z50
N17 G1 Z-5
N18 G145 X=E1+E5-E6 Y=E2 L0 E7
N19 G29 E0 E0=E7<1 N=49
N20 G145 X=E1+E5-E6-2 Y=E2 L1 X7=10
N21 G0 X=E1+E5-E6-2 Y=E2
N22 G3 I=E1 J=E2 X=E1 Y=E2+E5-E6-2
N23 G145 X=E1 Y=E2+E5-E6 L0 E7
N24 G29 E0 E0=E7<1 N=49
N25 G145 X=E1 Y=E2+E5-E6-2 L1 Y7=11
N26 G0 X=E1 Y=E2+E5-E6-2
N27 G3 I=E1 J=E2 X=E1-E5+E6+2 Y=E2
N28 G145 X=E1-E5+E6 Y=E2 L0 E7
N29 G29 E0 E0=E7<1 N=49
N30 G145 X=E1-E5+E6+2 Y=E2 L1 X7=12
N31 G0 X=E1-E5+E6+2 Y=E2
N32 G3 I=E1 J=E2 X=E1 Y=E2-E5+E6+2

```

```

N33 G145 X=E1 Y=E2-E5+E6 L0 E7
N34 G29 E0 E0=E7<1 N=49
N35 G145 X=E1 Y=E2-E5+E6+2 L1 Y7=13
N36 G0 X=E1 Y=E2-E5+E6+2
N37 G0 Z50
N38 M28

N39 E20=E1+(E10+E12):2 (measured X coordinate)
N40 E21=E2+(E11+E12):2 (measured Y coordinate)
N41 E22=sqrt((E10-E20)^2+(E2-E21)^2) (First radius hole)
N42 E22=E22+sqrt((E11-E21)^2+(E1-E20)^2) (Second radius hole)
N43 E22=E22+sqrt((E12-E20)^2+(E2-E21)^2) (Third radius hole)
N44 E22=(E22+sqrt((E13-E21)^2+(E1-E20)^2)):4 (Radius hole)
N45 G29 E1 E1=1 N=51

```

```

N46 M0
N47 (HOLE OUT OF TOLERANCE)
N48 G29 E0 E0=1 N=51
N49 M0
N50 M30

```

Explanation:

```

N1  : Define the plane of operation
N2  : Activate a stored aero offset
N3  : Load mill (d=20 mm)
N4-N10 : Edit e-parameters
N11 : Define drill cycle
N12 : Drill hole
N13 : Retract tool and stop spindle
N14 : Load probe
N15 : Activate probe
N16 : Move to the first hole
N17 : Move to depth
N18 : Measure first position with high feedrate
N19 : Measuring position reached, but no measurement was carried out. Jump to error message
      (E7=0)
N20 : Measure first measuring position with measurement feed (E10 is measuring position in X
      direction)
N21 : Return to starting position (first measuring position)
N22 : Move to the second hole with a circular movement
N23 : Measure second position with high feedrate
N24 : Measuring position reached, but no measurement was carried out. Jump to error message
      (E7=0)
N25 : Measure second measuring position with measurement feed (E11 is measuring position in Y
      direction)
N26 : Return to starting position (first measuring position)
N27 : Move to the third hole with a circular movement
N28 : Measure third position with high feedrate
N29 : Measuring position reached, but no measurement was carried out. Jump to error message
      (E7=0)
N30 : Measure third measuring position with measurement feed (E12 is measuring position in X
      direction)
N31 : Return to starting position (first measuring position)
N32 : Move to the third hole with a circular movement
N33 : Measure fourth position with high feedrate
N34 : Measuring position reached, but no measurement was carried out. Jump to error message
      (E7=0)

```

N35 : Measure fourth measuring position with measurement feed (E13 is measuring position in Y direction)
N36 : Return to starting position (first measuring position)
N37 : Retract probe to avoid collision
N38 : Deactivate probe
N39 : Calculate X centre point (E20)
N40 : Calculate Y centre point (E21)
N41 : Calculate first radius (Distance between measure position and the new centre point)
N42 : Add second radius
N43 : Add third radius
N44 : Add fourth radius and divide by four.
N45 : Jump to the end of the program
N46 : Stop program
N47 : Text
N48 : Jump to the end of the program
N49 : Stop program
N50 : Program end

61. Read probe status G148

Purpose

To read the probe status in measuring cycle macros.

Format

N... G148 E...

Parameters

E E-parameter for probe status

Associated Functions

G145, G149, G150

Type of function

Non modal

Notes and usage

PROBE STATUS

The E-parameter can have one of four values:

E... = 0: the probe is not activated; no contact with measuring surface.

E... = 1: the probe is activated; in contact with the measuring surface.

E... = 2: Block Search, Testrun or the Demo mode is active.

E... = 3: a probe error is active; no measurements can be made.

The priority for probe status codes is:

priority:

1: code 2 (active mode)

2: code 3 (probe error)

3: code 0 or 1 (probe contact)

INTERRUPT

The G148 function cannot be stopped by an interrupt command.

Example

N110 G148 E27

N115 G29 E91=E27=2 E91 N=300

Explanation:

N110: Store probe status in E-parameter number 27.

N115: Jump to block N300 if the program is executed in Block Search, Testrun or Demo. In this way e.g. calculation with parameters which are not loaded while no measurement was executed, can be avoided.

Note: The function G148 is not permitted when G182 is active.

Read probe status G148

62. Read tool data and zero offset G149

Purpose

To read tool data or zero offset values and store them in specified E-parameters within measuring cycle macros.

Format: Tool data

To read active tool number:

N... G149 T0 E...

To read tool dimensions:

N... G149 T... {T2=...} {L1=...} {R1=...} {M1=...}

To read tool status:

N... G149 T... E...

Format: Zero offsets

To read active zero offset G-functions:

N... G149 N1 =0/1 E...

To read stored zero offsets:

With standard zero offsets or MC84=0:

N... G149 N1=51...59 [(axis address)7=...] {(axis address)7=...}

With MC84>0 zero offsets extends:

N... G149 N1=54.[nr] [(axis address)7=...] {(axis address)7=...}{B47=...}

To read programmable zero offset:

N... G149 N1 =92 [(axis address)7=...] {(axis address)7=...}

Format: Actual position

To read the actual position of X,Y or Z:

N... G149 [(axis address)7=...] {(axis address)7=...}

Parameters

Tool data

T Tool number

T2= Tool offset index

E E-parameter

L1= E-parameter for toollength

R1= E-parameter for tool radius

M1= E-parameter for toollife

Zero offsets

N1= Zero offset shift

X7= E-par. for offset/position in X

Y7= E-par. for offset/position in Y

Z7= E-par. for offset/position in Z

A7= E-par. for offset/position in A

B7= E-par. for offset/position in B

C7= E-par. for offset/position in C

B47= E-parameter for rotation in B4=

Associated Functions

G145, G148, G150

Type of function

Non-modal.

Notes and usage

TOOL NUMBER (T)

Number of tool for which tool data must be read. If the FMS-tool memory (Flexible Manufacturing System) is in use, the complete number including the spare tool index has to be written.

TOOL DATA

The toolradius (R1=..), toollength (L1=..) and the remaining toollife (M1=..) can be read.

TOOL OFFSET INDEX (T2=)

A tool offset index 0,1 or 2 can be specified. Default is T2=0.

TOOL STATUS (E)

The tool status from the tool memory will be loaded in the indicated E-parameter. The tool status can have the following values.

E... = 1: the tool is enabled and measured.

E... = 0: the tool is enabled but not measured

E... = -1: the tool is disabled

E... = -2: the tool time is expired

E... = -4: tool breakage error

E... = -8: the tool cutting force is exceeded

E... = -16: the tool time < T3 = programmed

A combination of error messages is also possible:

E... = -13 means: error message -8 and -4 and -2 and 1.

READING ADDRESSES WITHOUT VALUE:

If addresses are read from the tool memory when they are not entered previously, a value of zero will be returned.

ZERO OFFSET NUMBER (N1=)

The number of the zero offset of which data has to be read. 'N1=' can have a value from 51 to 59, **54.[nr]** or delete 92. Reading G92 offsets when G92 is not active will result in offset values of zero.

ZERO OFFSET GROUP (N1=)

The zero offset group of which the active G-function has to be read. N1= can have the value 0 or 1.

N1=0: If G52 is active, the E-parameter is given the value 52. If G52 is not active, the E-parameter is given the value 51.

N1=1: The E-parameter is given the value of the active offset G54 - G59. IF a G54-G59 type offset is not active, the E-parameter is given the value 53.

INTERRUPT

The function cannot be stopped by an interrupt command.

Note: The function G149 is not permitted when G182 is active.

The tool data of T0 can not be read. If T0 is used, the relevant E-parameters are not loaded. No error message is given to this effect.

Examples

EXAMPLE 1: to read the active tool number.

N100 G149 T0 E1

E1 contains the number of the active tool

EXAMPLE 2: to read the active tool dimensions.

N100 G149 T=E1 L1=5 R1=6

Read the tool dimensions of tool T=E1

E5 contains the tool length

E6 contains the tool radius

EXAMPLE 3: to read the active tool dimensions.

N100 G149 T12 L1=5 R1=6

Read the tool dimensions of tool T12

E5 contains the tool length

E6 contains the tool radius

EXAMPLE 4: to read the active zero offset function

N100 G149 N1=0 E2

N110 G149 N1=1 E3

E2 contains the active preset function (51 or 52)

E3 contains the active zero offset function (53 to 59) or **G54.[nr]**

EXAMPLE 5: to read a stored zero offset.

N100 G149 **N1=54** X7=1 Z7=2

or

N100 G149 **N1=54.[nr]** X7=1 Z7=2

Read offset G54.

E1 contains X-axis offset.

E2 contains Z-axis offset.

EXAMPLE 6 : Calling a shift with angle of rotation of coordinate system

N100 G149 N1=54.02 X7=1 B47=2

Call shift G54.02

E1 has shift in X

E2 has angle of rotation of coordinate system

Example 7 : Calling a zero point shift (G92/G93)

N100 G149 N1=92 X7=1 Z7=2

Call shift G92

E1 has shift in X

E2 has shift in Z

If a G92/G93 shift is called while G92/G93 is ineffective, the shift values 0 are obtained.

63. Write tool data and zero offset G150

Purpose

To write values in the Tool memory or Zero offset memory within measuring cycle macros.

Format: Tool data

To write data in Tool memory

N... G150 T... {T2=...} {L1=...} {R1=...} {M1=...}

To write tool status in Tool memory:

N... G150 T... E...

Zero offsets

To write data in zero offset memory:

With standard zero offsets or MC84=0:

N... G150 N1=54...59 [(axis address)7=...] {(axis address)7=...} etc.

With MC84>0 zero offsets extends:

N... G150 N1=54.[nr] [(axis address)7=...] {(axis address)7=...} etc.

Parameter :

Tool data

T Tool number

T2= Tool offset index

E E-parameter

L1= Toollength value in T

R1= Tool radius value in T

M1= Toollife value in T

Zero offsets

N1= Zero offset shift

X7= Offset in X

Y7= Offset in Y

Z7= Offset in Z

A7= Offset in A

B7= Offset in B

C7= Offset in C

B47= Angle of rotation in B4=

Associated Functions

G145, G148, G149

Type of function

Non modal

Notes and usage

TOOL NUMBER (T)

Number of tool for which tool data must be changed. If the FMS-tool memory (Flexible Manufacturing System) is in use, the complete number including the spare tool index has to be written.

The modal number of the actual tool is not influenced by this command.

TOOL OFFSET INDEX (T2=)

A tool offset index 0,1 or 2 can be specified. Default is T2=0. The offset index of the actual tool is not influenced by this command.

TOOL STATUS (E)

The tool status can be loaded from the indicated E-parameter into the tool memory.

Possible values for tool status are:

E... = 1: the tool is enabled and measured.

E... = 0: the tool is enabled but not measured

E... = -1: the tool is disabled

E... = -2: the tool time is expired

E... = -4: tool breakage error

E... = -8: the tool cutting force is exceeded

E... = -16: the tool time < T3 = programmed

A combination of error messages is also possible:

E... = -13 means: error message -8 and -4 and -2 and 1.

ZERO OFFSET (N1=)

Zero offset (G52, G54-G59 or **G54.[nr]**) that has to be changed.

INTERRUPT

The function cannot be stopped by an interrupt command.

Note: The function G150 is not permitted when G182 is active.
The tool data of T0 can not be loaded.

Examples

EXAMPLE 1. Write data in the tool memory.

N50 G150 T1 L1=E2 R1=4

Explanation:

Change the data of tool No.1.

Store the value of parameter E2 as the tool length.

Make the tool radius equal to 4.

EXAMPLE 2. Write data in the zero offset memory.

N70 G150 **N1=57** X7=E1 Z7=E6

or

N70 G150 **N1=54.3** X7=E1 Z7=E6

Explanation:

Zero offset values of G57 are to be changed.
Store value of parameter E1 in G57 or **G54.3** X-axis offset
Store value of parameter E6 in G57 or **G54.3** Z-axis offset

Example 3 : Changing a zero point shift with angle of rotation of coordinate system:

N70 G150 **N1=54.03** X7=E1 B47=E6

Explanation:

Change zero point shift values of G54.03
Store value of E1 parameter in **G54.3** shift in X
Store value of E6 parameter in **G54.3** shift in B4=

Write tool data and zero offset G150

64. Basic coordinate system G180

Purpose

To select the cylindrical coordinate system which allows an easy way of programming contours and positions on the curved surface of the cylinder.

Formats

G180: Basic coordinate system

N... G180 [auxiliary axis 1][auxiliary axis 2][Tool axis]

Parameters

X,Y,Z Cylinder plane:2 / Tool axis:3

A,B,C Cylinder plane:1

R Cylinder radius

General principles

The normal expression is G180 X1 Y1 Z1

The only following configuration are possible:

Auxiliary axis 1 X

Auxiliary axis 2 Y

Tool axis Z or W

The correct procedure depends on 3 different types of information:

- 1) The tool axis is determined by G17/G18/G19 (G17 Z).
- 2) G180 determines which axes are to be substituted. (G17 W in Z)
- 3) The machine constants for the tool axis definition should also be correct (tool axis W belongs to Z).

Associated functions

None

Type of function

Modal

Notes and usage

FUNCTIONS TO BE CANCELLED

The functions G41-G44, G64, G73, axis rotation (G92/G93 B4=) and G141 must be cancelled before G180 is activated.

Any other function which is active immediately before the G180 block remains active.

Note: The words X,Y,Z cannot be programmed without any value. Therefore the value 1 is written to it. This value has no meaning.

RADIUS AND TOOL LENGTH COMPENSATION

The tool length compensation is active in the defined tool axis. The radius compensation is active in the auxiliary axis.

MACHINE CONSTANT

The machine constant must be accurate. If W-Axis is the fourth axis then MC117 = 3 (just like Z-Axis). MC3401 = 0 (W-Axis is a Linear axis).

COORDINATES

Only cartesian coordinates can be used.

Note: If G180 is programmed and radius compensation is still active, the compensation is cancelled with the G180. It is advised to cancel radius compensation with G40 and then to return to the basic coordinate system.

CANCELLATION

Cylinder interpolation is cancelled by either the G180 function or CLEAR CONTROL.

DEFAULT MODE

When turning on the controller or activating Softkey CLEAR CONTROL G180 X1 Y1 Z1 is automatically turned on.

Example

EXAMPLE 1

```
N12340
N1 G17 S1000 T1 M6
N2 G54
N3 G180 X1 Y1 W1
N4 G81 Y2 B10 Z-22 F1000 M3
N5 G79 X0 Y0 Z0
```

Explanation:

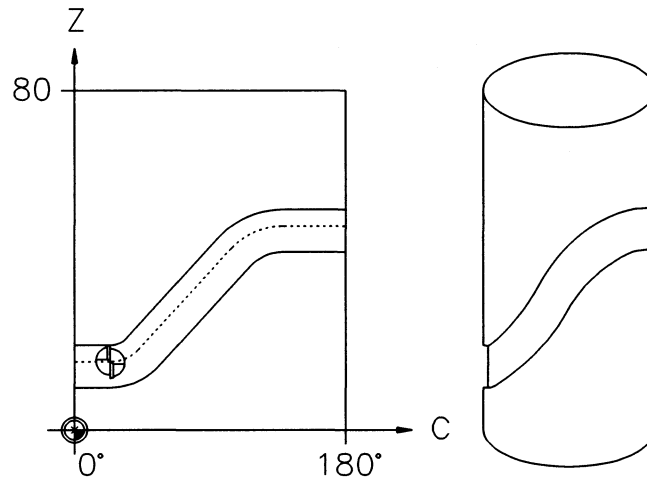
- N1: Define plane of operation.
- N2: Activate a stored zero offset.
- N3: Activate auxiliary plane XY and Tool axis W.
- N4: Define drilling cycle.
- N5: Drilling hole with the feedrate active in the W axis.

65. Cylindrical coordinate system G182

Purpose

Selection of the cylindrical coordinate system. Using this system, you can easily program contours and positions on the curved surface of a cylinder.

Formats



G182: Cylindrical coordinate system

To activate the cylindrical coordinate system
N.. G182 [cylinder axis] [rotary axis] [tool axis] R..

N.. G182 R..

Parameters

X,Y,Z Cylinder plane:2 / Tool axis:3
A,B,C Cylinder plane:1
R Cylinder radius

With movements

X, Y, Z, U, V, W Linear axis coordinate
A, B, C Rotary axis coordinate
F Feedrate along the cylinder surface

Modal words

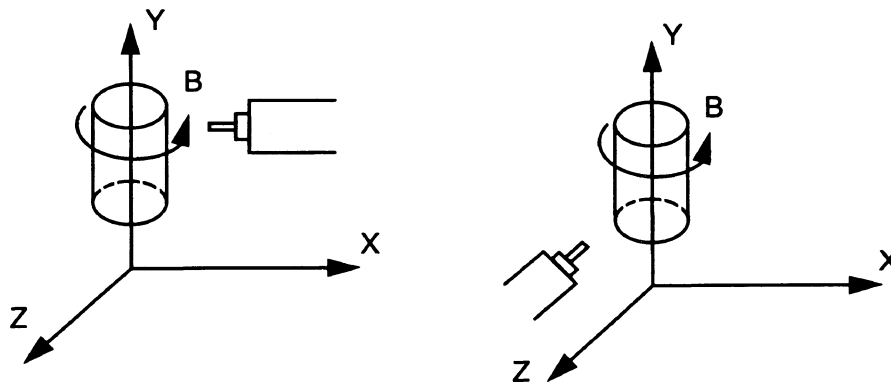
S,T,H,M,E,F1=,T1=, T2=

General principles

Contours on the curved surface of a cylinder are drawn in a plane representing the curved surface. This plane is defined with the rotary axis, the cylinder axis and the cylinder radius. In this plane linear and circular movements with radius compensation can be programmed. During the execution of the program these movements are converted to movements with a linear (= the axis of the cylinder) and a rotary axis (= the axis rotated about the cylinder axis). This is called cylinder interpolation.

In general the following configurations are possible:

Rotary axes	A,	B,	C
Cylinder axes	X,	Y,	Z
Toolaxis	Y or Z,	X or Z,	X or Y.



NB9661

The BY-plane for cylinder interpolation

The correct procedure depends on 3 different types of information:

- 1) The tool axis is determined by G17/G18/G19 (G17 Z).
- 2) G182 determines which axes are to be substituted. (G17 AX or BY)
- 3) The machine constants for the rotary axis definition should also be correct. (rotary axis A belongs to X).

Associated functions

None

Type of function

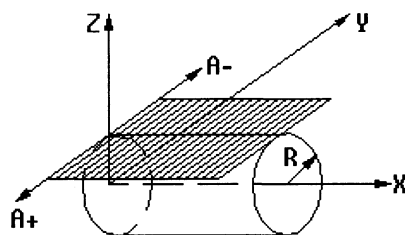
Modal

Notes and usage**FUNCTIONS TO BE CANCELLED**

The functions G41-G44, G64, G73, axis rotation (G92/G93 B4=) and G141 must be cancelled before G182 is activated.

Any other function which is active immediately before the G182 block remains active.

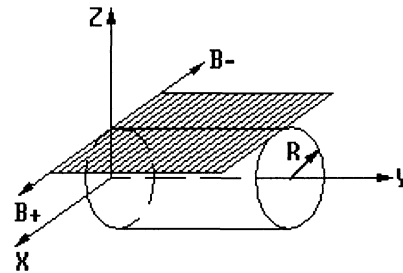
SPECIFYING THE CYLINDER PLANE



G182 A1 X2 Z3 R

1/4

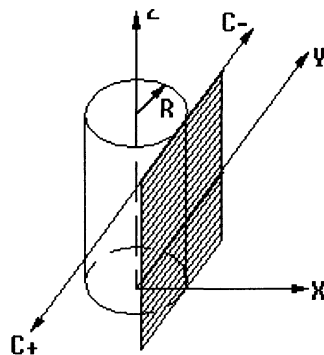
G182 A1 X2 Z3 R..
or (as previous)
G182 A1 X1 Z1 R..



G182 B1 Y2 Z3 R

2/4

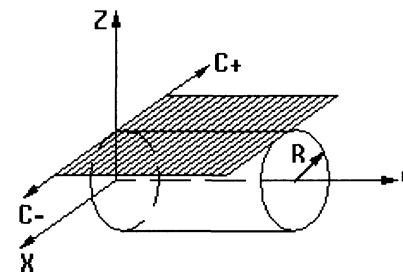
G182 B1 Y2 Z3 R..
or (as previous)
G182 B1 Y1 Z1 R..



G182 C1 Z2 X3 R

3/4

G182 C1 Z2 X3 R..
or (as previous)
G182 C1 X1 Z1 R..



G182 C1 Y2 Z3 R

4/4

G182 C1 Y2 Z3 R..

Note: The words X, Y, Z, A, B, C cannot be programmed without any value.

In the G182-block the configuration for cylinder interpolation is programmed:

(standard configuration)

Rotary axis:	A1	B1	C1
Cylinder axis:	X1	Y1	Z1
Tool axis:	Y1/Z1	X1/Z1	X1/Y1
Cylinder radius:	R	R	R

(other configuration)

Rotary axis marked with 1:	A1	B1	C1
Cylinder axis marked with 2:	X2Y2Z2	Y2X2Z2	Z2X2Y2
Tool axis marked with 3:	Y3Z3X3	X3Z3Y3	X3Y3Z3
Cylinder radius:	R	R	R

Note: When the mark is 1 then only a standard configuration is possible.

MACHINE CONSTANTS

The machine constants must be accurate.

MC 102 = 1, MC103 = 88 (X-Axis)

MC 107 = 2, MC108 = 89 (Y-Axis)

MC 112 = 3, MC113 = 90 (Z-Axis)

MC 117 = 4 belongs to axis 1 (4-3), MC118 = 65 (A-Axis turns around X-Axis)

MC 122 = 6 belongs to axis 3 (6-3), MC123 = 67 (C-Axis turns around Z-Axis)

DEFAULT CYLINDER PLANE

When a machine tool has only one rotary table, the configuration for cylinder interpolation is defined in the Machine Constants. Therefore, if the axis configuration is not programmed in a G182-block, these settings are used automatically by the CNC.

CYLINDER RADIUS

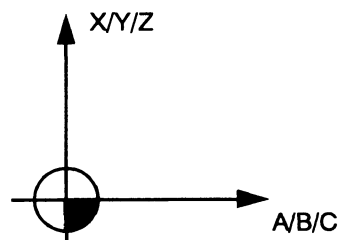
The cylinder radius is used by the CNC for calculating the feedrate of the rotary axis. The radius value must be between 5 mm and 500 mm. If the R-word is not programmed in a G182-block, an error message is displayed.

CHANGING THE CYLINDER RADIUS

The radius of the cylinder can be changed with the R-word in another G182-block. In this block the definition of the plane has to be repeated.

PLANE OF THE CURVED SURFACE

The plane of the curved surface is called either AX, BY or CZ, depending on which cylinder the contour has to be made.



NB9656

Axes of the plane of the curved surface

The horizontal axis is the rotary axis and is programmed with the corresponding axis address A, B, C in degrees and decimal parts thereof.

The vertical axis is the cylinder axis and is programmed with the corresponding axis address X, Y, Z in mm or inches.

The tool axis is perpendicular to the wall of the cylinder, programmed in mm or inches with the addresses Y or Z for the AX-plane, X or Z for the BY-plane, X or Y for the CZ-plane depending on which axis the tool is loaded.

DATUM POINT

The datum point in the rotary, cylinder and tool axis must be programmed, before cylinder interpolation is activated. This can be achieved with

G51-G52 (p) reset axes.

G53-G59 or G54I[nr.] stored zero offset

G92/G93 a datum point shift

Once cylinder interpolation (G182) is activated a datum point shift is not allowed until the basic coordinate system (G180) is chosen again.

COORDINATES

Only cartesian coordinates can be used.

The functions G90 and G91 are used for programming absolute (G90) or incremental (G91) dimensions and can be used with the rotary, cylinder and tool axis.

RAPID MOVEMENTS (G0)

A rapid traverse movement is programmed with G0 and the end point of the movement. Two or three axes can be programmed in one block. The axes move with the positioning logic:

Tool towards the cylinder: 1. movement in the plane
2. tool axis movement

Tool from the cylinder: 1. tool axis movement
2. movement in the plane

LINEAR FEED MOVEMENTS (G1)

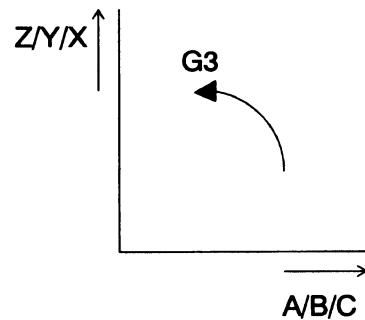
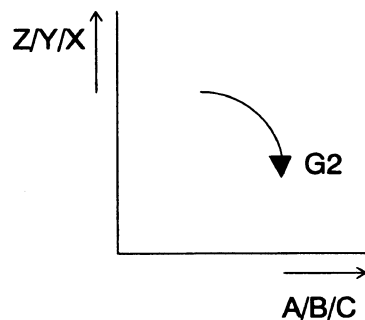
A linear feed movement is programmed with G1, the end point of the movement and the feedrate.

Two or three axes can be programmed in one block. All axes move simultaneously and reach their end point at the same time.

The programmed feed rate is the surface feed on the cylinder at the radius from the G182-block.

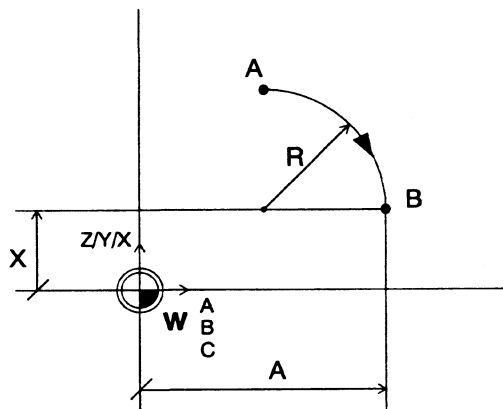
CIRCULAR FEED MOVEMENTS (G2/G3)

A circular feed movement can only be programmed with a G2 or G3, the end point coordinates and the radius of the arc (R-word).

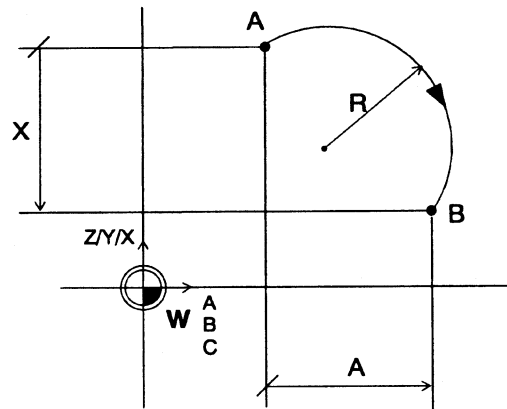


NB7970

Direction of circular movement



G90



G91

NB7971

Circular arc with end point and radius

RADIUS COMPENSATION

For radius compensation in the plane of the cylinder the functions G40, G41, G42, G43 and G44 can be used. These functions have the same meaning as in the basic coordinate system (G180 active). For defining LEFT and RIGHT one should look from the tool to the cylinder.

Note: If G180 is programmed and radius compensation is still active, the compensation is cancelled with the G180. It is advised to cancel radius compensation with G40 and then to return to the basic coordinate system.

TOOL SIZE

From the control side of view there is no restriction on the size of the tool radius. However, if the tool radius is too large, undercuts may be produced. These undercuts depend on the shape and size of the tool and the depth of operation.

Note: With contours on the cylinder the greatest accuracy is achieved with a tool of which the diameter is about 0.2 mm less than the width of the groove.

CANCELLATION

Cylinder interpolation is cancelled by either the G180 function or CLEAR CONTROL.

DEFAULT MODE

G180 is made active automatically when the CNC is switched on, or the CLEAR CONTROL operation is performed.

FUNCTIONS PERMITTED

G0, G1, G2/G3, G4, G14, G22, G23, G29, G40-G44, G90/G91, G94/G95, G180/G182

Notes: 1. If G14 or G29 are used during cylinder interpolation, the target block number for the jump, must be in the program section for cylinder interpolation.

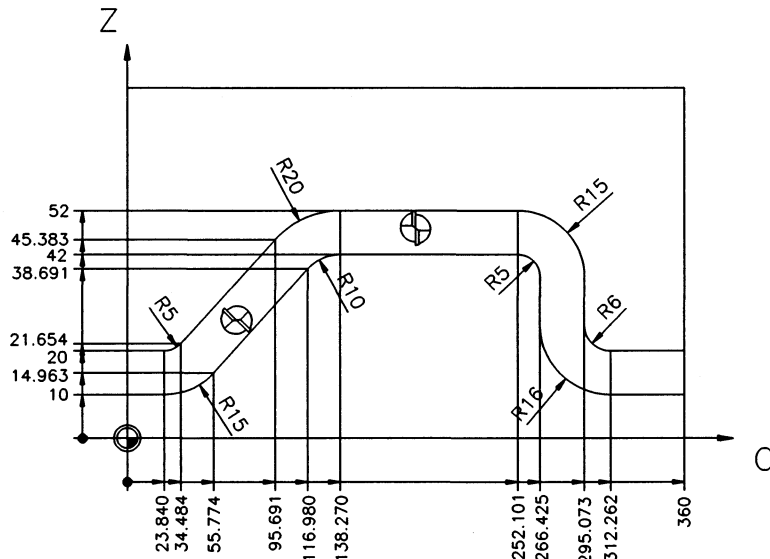
2. The function G94 or G95 which is active at activating cylinder interpolation, is not influenced by G182.

3. All other G-functions are not permitted, when G182 is active.

Example**EXAMPLE 1.**

The groove on the curved surface of a cylinder of diameter=40 mm has to be milled with a slotting end mill $D=9.5$ mm. The depth of operation=4 mm. The workpiece is machined horizontal with the rotary axis C, the cylinder axis Z and tool axis Y.

The contour is programmed in this example.



```

N12340
N1 G18 S1000 T1 M6
N2 G54
N3 G182 Y1 C1 Z1 R20
N4 G0 Z15 C0 Y22 M3
N5 G1 Y16 F200
N6 G43 Z10
N7 G41
N8 G1 C23.84
N9 G3 Z14.963 C55.774 R15
N10 G1 Z38.691 C116.98
N11 G2 Z42 C138.27 R10
N12 G1 C252.101
N13 G2 Z37 C266.425 R5
N14 G1 Z26
N15 G3 Z10 C312.262 R16
N16 G1 C365
N17 G40
N18 G41 Z20
N19 G1 C312.262
N20 G2 Z26 C295.073 R6
N21 G1 Z37
N22 G3 Z52 C252.101 R15
N23 G1 C138.27
N24 G3 Z45.383 C95.691 R20
N25 G1 Z21.654 C34.484
N26 G2 Z20 C23.84 R5
N27 G1 C0
N28 G40
N29 G180
N30 G0 Y100 M30

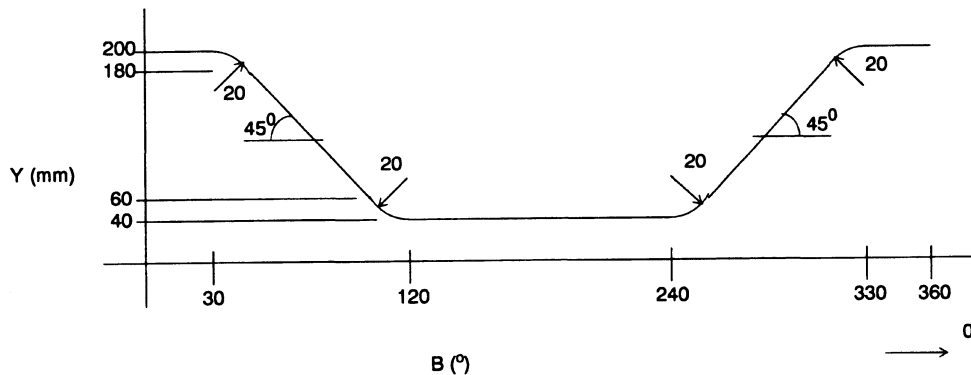
```

Explanation

- N1: Load the tool
- N2: Activate the stored zero offset of G54
- N3: Use the cylindrical coordinate system for the BY-plane, the Y-axis as tool axis and a cylinder radius of 20 mm.
- N4: Move to the start position and activate the spindle
- N5: Feed movement to depth
- N6: Radius compensation T0 the lower contour
- N7: Activate radius compensation LEFT
- N8-16: Points of the lower contour till 365° (=360°+run out)
- N17: Cancel radius compensation; the tool tip should come at the position 365°.
- N18: Activate radius compensation LEFT and move to upper contour
- N19-27: Points of the upper contour
- N28: Cancel radius compensation
- N29: Return to the basic coordinate system of the machine tool
- N30: Retract tool from part and end of program.

EXAMPLE 2.

The upper part of a groove on the curved surface of a cylinder with a radius of 114.6 mm is drawn. The workpiece is machined horizontal with the rotary axis B, the cylinder axis Y and tool axis Z. The contour is programmed in this example.



NB8558

```

N9011
N1 G17
N2 G54
N3 S500 T1 M6
N4 G182 Y1 B1 Z1 R114.6
N5 G0 Y180 B0 Z116 M3
N6 G43 Y200
N7 G1 Z114 F300
N8 G42
N9 B30
N10 G2 Y194.142 B37.071 R20
N11 G1 Y45.858 B112.929
N12 G3 Y40 B120 R20
N13 G1 B240
N14 G3 Y45.858 B247.071 R20
N15 G1 Y194.142 B322.929
N16 G2 Y200 B330 R20
N17 G1 B360
N18 G40
N19 G0 Z150 M30

```

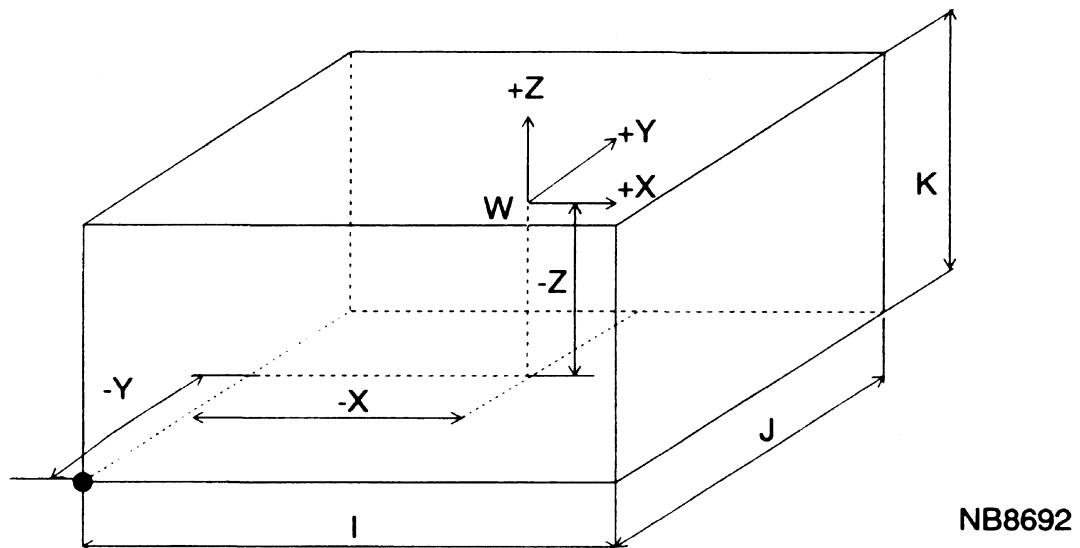
Explanation

- N1: Set the plane for operation
- N2: Activate the stored zero offset from G54
- N3: Load tool 1 and activate the spindle
- N4: Use the cylindrical coordinate system for the BY-plane, the Z-axis as tool axis and a cylinder radius of 114.6 mm.
- N5: Rapid tool movement to start position and start the spindle
- N6: Move tool to the contour
- N7: Feed movement at depth
- N8: Set radius compensation RIGHT
- N9-17: Move along the contour. The contour points must be calculated by the part programmer from the data given on the drawing.
- N18: Cancel radius compensation
- N19: Retract tool from part and end of program.

66. Graphic window definition G195

Purpose

To define the dimensions of a 3D graphic window and its position relative to the zero point W. In this window the workpiece and machine parts can be situated for a graphical simulation of a part program run.



NB8692

3D graphic window definition

Format

N... G195 X... Y... Z... I... K... {B...} {B1=...} {B2=...}

Parameters

X,Y,Z Start point coordinate
 I Dimension parallel to X
 J Dimension parallel to Y
 K Dimension parallel to Z
 B Rotation around hor. axis (3D)
 B1= Rotation around vert. axis (3D)
 B2= Rotation around third axis (3D)

Associated functions

G98, G99, G196 to G199

Type of function

Non-modal.

Notes and usage

GRAPHICAL SUPPORT

Refer to the appendix GRAPHICAL SUPPORT at the end of this manual for a short overview about the graphical support provided in the CNC PILOT control system and to the user manual for using the graphical support.

GRAPHIC WINDOW

The window, thus a bounded area on the display, is a rectangular 3D box which dimensions are defined by the G195-function.

The window is used with the graphical simulation, but also with the synchron graphics with which the actual tool movements on the machine can simultaneously be seen on the display of the control.

RELEVANT WINDOW AXIS

Because the display on the control is a rectangle, the scale on the shorter side (Y-axis in XY-plane) which is calculated from the programmed value (the J-word), also determines the scale on the longer axis (X-axis in XY-plane).

CONTOUR DEFINITION (G196 - G199)

Besides a window also an outer contour of a workpiece blank and/or machine parts and, if required, an inner contour can be defined for the graphical simulation. The dimensions of these contours are programmed with the functions G196 to G199. Refer to these functions for defining a contour.

DEFAULT WINDOW DIMENSIONS

If the dimensions of the 3D window are not defined the CNC uses the software limit switches' distances as default values.

ANGLE OF VIEWING (B, B1=, B2=)

With the synchron graphics or the 3D-wire plot the workpiece can be seen rotated. The angles for viewing the rotated workpiece on the display are defined by the words B, B1= or B2=.

	B rotation about	B1= rotation about	B2= rotation about
XY-plane (G17)	X-axis	Y-axis	Z-axis
XZ-plane (G18)	Z-axis	X-axis	Y-axis
YZ-plane (G19)	Y-axis	Z-axis	X-axis

Other methods are available for selecting an angle of viewing and are described in the user manual.

DEFAULT SETTINGS FOR ANGLES OF VIEWING

If the angles of viewing are not programmed the following default settings are automatically used by the control:

B60, B1=30 and B2=0

RESTRICTIONS

The function G195 is not permitted in MDI or the TEACH-IN (PLAYBACK) mode.

Example

```
N9000
N1 G17
N2 G195 X-30 Y-30 Z-70 I170 J150 K100
N3 G199
```

Explanation:

N1: Define the machining plane.
 N2: Define the graphic window.
 N3: Start of the contour description section

67. End contour description G196

Purpose

To end the contour description for the graphical simulation of a partprogram run.

Format

N... G196

Associated functions

G98, G99, G195, G197 to G199

Type of function

Non-modal.

Notes and usage

FUNCTIONS ACTIVE AFTER A G196 BLOCK

After a G196 the function G64 is reset, so G63 is activated again.

If the last function of the contour description is a G2 or G3, this function is set back to the active function G0 or G1 before the G199.

All other modal functions which are active before the G199, are not influenced.

RESTRICTIONS

1. A G199 must be programmed before the G196. If not, an error message is displayed.
2. The G196 function cannot be used in MDI and the TEACH-IN (PLAYBACK) mode.

Example

```
N9000
N1 G17
N2 G195 X-30 Y-30 Z-70 I170 J150 K100
N3 G199 X0 Y0 Z0 D-20
...
N10 G196
```

Explanation

```
N1 : Define the plane of operation
N2 : Define Graphic window
N3 : Start Graphic contour description
:
N10 : End Graphic contour description.
```

End contour description G196

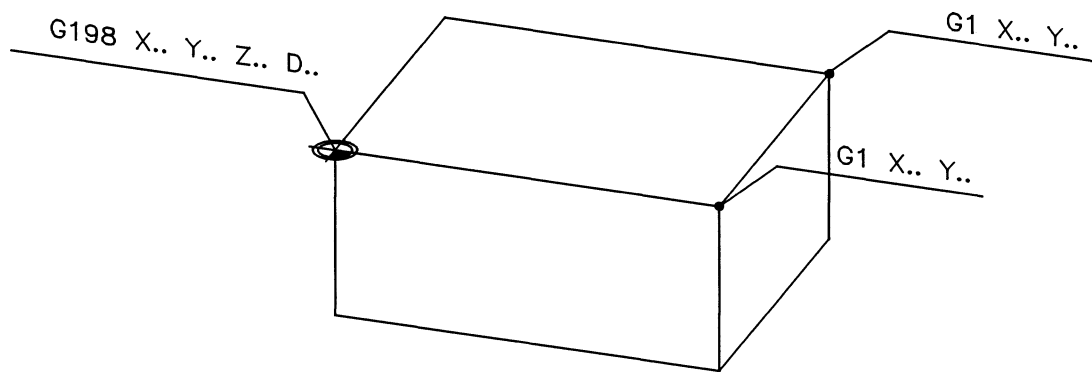
68. Begin inner/outer contour description G197/G198

Purpose

To define the start point in a graphical presentation of a contour from a blank workpiece or a machine part. Outer as well as inner contours can be defined. After defining the start point of the contour the contour itself can be programmed with the functions G1 and G2 or G3.

G197: define the start point of an inner contour

G198: define the start point of an outer contour



Start point for defining the blank contour.
0: zero point defined by G199.

Format

To define the start point of an inner contour

N... G197 X... Y... D...

To define the start point of an outer contour

N... G198 X... Y... {Z...} D...

Parameters

X,Y,Z Startpoint coordinate

D Depth inside contour

Associated functions

G98, G99, G195 to G196, G199

Type of function

Non-modal.

Notes and usage

START POINT OF CONTOUR DESCRIPTION

The start point of the contour is programmed with the linear axes coordinates X, Y and Z. These coordinates are related to the point programmed in a G199 block.

The coordinates of the start point must be absolute, cartesian coordinates. Polar coordinates are not allowed.

E-parameter can be used with the functions G197/G198, but parameter definition is not allowed in the G197/G198 block itself.

If the Z-word is not programmed in a G198 block, Z0 is used as a default setting.

Note: The Z-word is not used with the G197.

CONTOUR DESCRIPTION

Once the start point of the contour is established (in a G197 or G198), the contour is programmed with the functions G1, G2 and G3.

G1:
a line with its end point.

G2/G3:
a circular arc with end point and radius or centre point and end point.

a complete circle with centre point only.

using the helix or 2.5 D interpolation is not allowed.

An end point or centre point can be programmed with absolute cartesian or polar coordinates. They are related to the point programmed in the G199 block.

For complicated contours the geometry of the control (G64) can be used.

The contour must be closed, otherwise, a straight line will be generated automatically from the end point to the start point.

The contour must lie in the main plane defined by the active function G17, G18 or G19.

THE DEPTH OF THE CONTOUR (D)

The depth of the outer contour is programmed with the D-word. Its value is related to the tool axis coordinate of the G198 block.

If the bottom of the contour is Z0 (XY-plane), the depth is positive. If the upper surface is Z0, the depth is negative.

The depth of the inner contour is also programmed with the D-word. Its value is related to the depth of the outer contour.

MORE THAN ONE CONTOUR DESCRIPTION

if a part is composed of independent contours, eg. layers or holes, it is possible to define each contour separately with the functions G198 and G197. With one of these functions the previous contour description is ended and the description of the next contour starts.

A complete contour, thus the outer as well as the inner contour, must be programmed in one section.

If all contours are defined the function G196 is used to end the complete contour description.

INNER CONTOURS

An inner contour must lie within a previously defined outer contour.

Inner contours may not intersect the sides or be tangent to the sides of the outer contour.

An inner contour cannot be inside another inner contour.

ENDING A CONTOUR DESCRIPTION

A contour description is finished with either a function G198 for defining another outer contour, a G197 for defining an inner contour or a G196 indicating that no more contours are followed.

CONTOUR DESCRIPTION IN MACRO

If a macro is to be used for describing a contour, eg. for describing machine parts, all graphic functions (G199, G198, G197 and G196) must be programmed in the macro.

FUNCTIONS ALLOWED WITH A CONTOUR DESCRIPTION

With a contour description only the following functions are permitted:

- G1, G2, G3: for describing a contour
- G64/G63: for defining the contour with the geometry function
- G196: for ending the contour description
- G197/G198: for defining another contour

FUNCTIONS IGNORED WITH THE CONTOUR DESCRIPTION

If radius compensation, scaling, mirror image or axis rotation is activated before the G199, the function is ignored during the graphical simulation of the contour. No error message is displayed to this effect.

Therefore it is advised to activate the following functions before the G199:

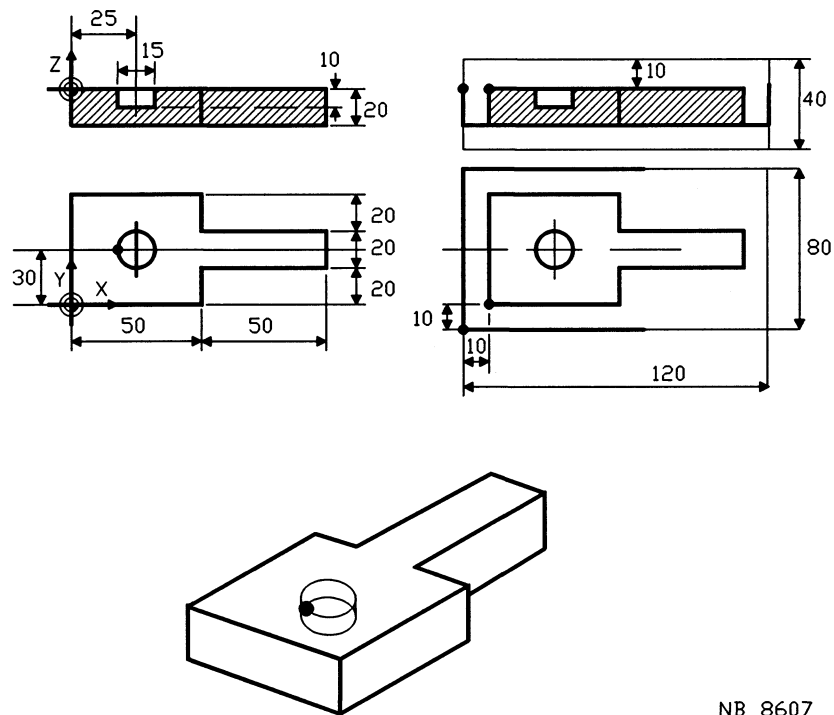
- G40 (no radius compensation)
- G72 (no mirror image or scaling)
- G90 absolute coordinates
- G93 B4=0 (no axis rotation)
- G180 Basic coordinate system

RESTRICTIONS WITH THE CONTOUR DESCRIPTION

1. The axes coordinates must be absolute and lie in the active main plane.
2. Polar coordinates and the combination of one coordinate and an angle can be used with the G1 and G2/G3 blocks. They cannot be used to define the start point of the contour.
3. Previously defined points cannot be used with a contour description.
4. The use of E-parameters to define the start point is not allowed. However, parameters can be used with the contour description itself (G1 and G2/G3 blocks).

RESTRICTIONS WITH THE USE OF G197 AND G198

1. With G197 the Z-coordinate is not used. The depth value (D-word) of the G197 is related to the depth of the outer contour.
2. The functions G197 and G198 are not permitted in MDI or in the TEACH-IN (PLAYBACK) mode.

Example**EXAMPLE 1: Outer and inner contour definition**

NB 8607

```

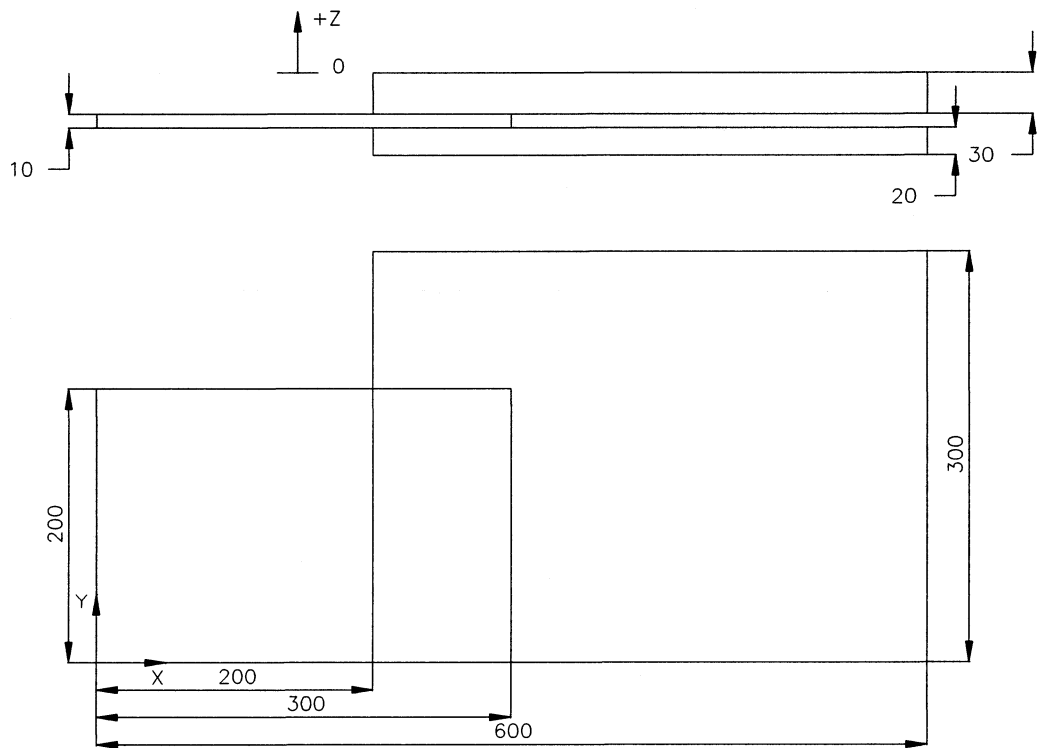
N1971981
N1 G17
N2 G195 X-10 Y-10 Z10 I120 J120 K-40
N3 G199 X0 Y0 Z0 B1 C2
N4 G198 X0 Y0 Z0 D-20
N5 G1 X50
N6 Y20
N7 X100
N8 Y40
N9 X50
N10 Y60
N11 X0
N12 Y0
N13 G197 X17.5 Y30 D-10
N14 G2 I25 J30
N15 G196

```

Explanation:

N1: Set the plane of operation to be the XY-plane
 N2: Set the graphic window to define the 3D space
 N3: Start of contour description section.
 N4: Define start point of outer contour.
 N5-N12: Description of the outer contour. The coordinates are related to the point programmed in the G199 block.
 N13: Define start point of inner contour.
 N14: Describe the inner contour to be a complete circle.
 N15: End of contour description section

EXAMPLE 2: Contour with different layers



NB9807

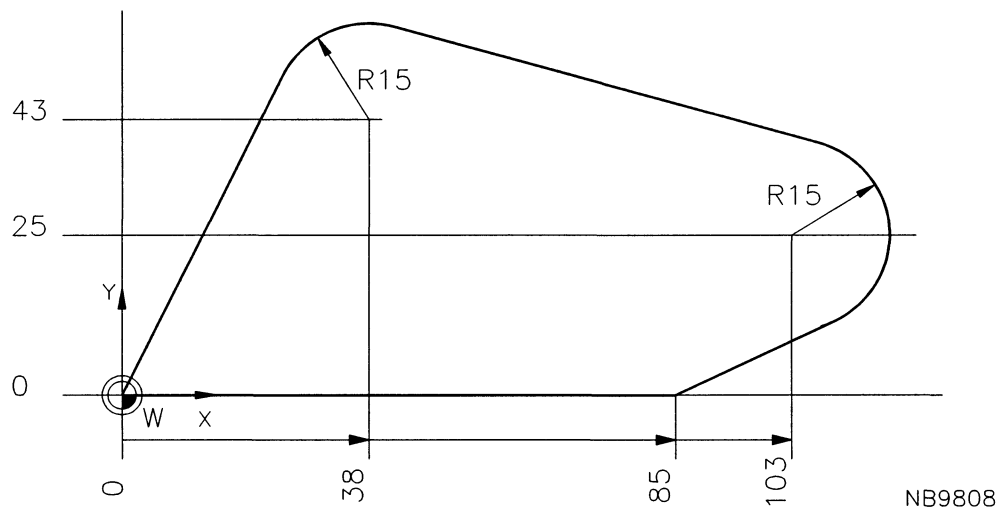
```

N1971982
N1 G54
N2 G17
N3 G195 X-5 Y-5 Z35 I610 J310 K-70
N4 G199 X0 Y0 Z0 B1 C2
N5 G198 X0 Y0 Z-30 D-10
N6 G1 X300
N7 Y200
N8 X0
N9 Y0
N10 G198 X200 Y0 Z0 D-30
N11 G1 X600
N12 Y300
N13 X200
N14 Y0
N15 G198 X200 Y0 Z-40 D-20
N16 G1 X600
N17 Y300
N18 X200
N19 Y0
N20 G196
:
    
```


Explanation:

N1: Set the program zero point
 N2: Set the plane of operation to be the XY-plane
 N3: Set the graphic window to define the 3D space
 N4: Start of contour description section.
 N5: Define start point of outer contour (first layer).
 N6-N9: Describe the first layer
 N10: Define start point of outer contour (second layer).
 N11-N14: Describe the second layer
 N15: Define start point of outer contour (third layer).
 N16-N19: Describe the third layer
 N20: End of contour description section

EXAMPLE 3: Contour described with geometry



N1971983
 N1 G54
 N2 G17
 N3 G195 X-10 Y-20 Z0 1140 J90 K-25
 N4 G199 X0 Y0 Z0 B1 C2
 N5 G198 X0 Y0 Z0 D-10
 N6 G64
 N7 G1 X85 Y0
 N8 R1=0
 N9 G3 I103 J25 R15 R1=0
 N10 G1 R1=0
 N11 G3 I38 J43 R15 R1=0
 N12 G1 X0 Y0
 N13 G63
 N14 G196
 N15 S1000 T1 M6
 N16 G0 X-15 Y-15 Z0 M3
 N17 G1 Z-10 F1000
 N18 G43 Y0
 N19 G42
 N20 G14 N1=6 N2=13
 N21 G40
 N22 G0 X-15 Y-15 Z100 M30

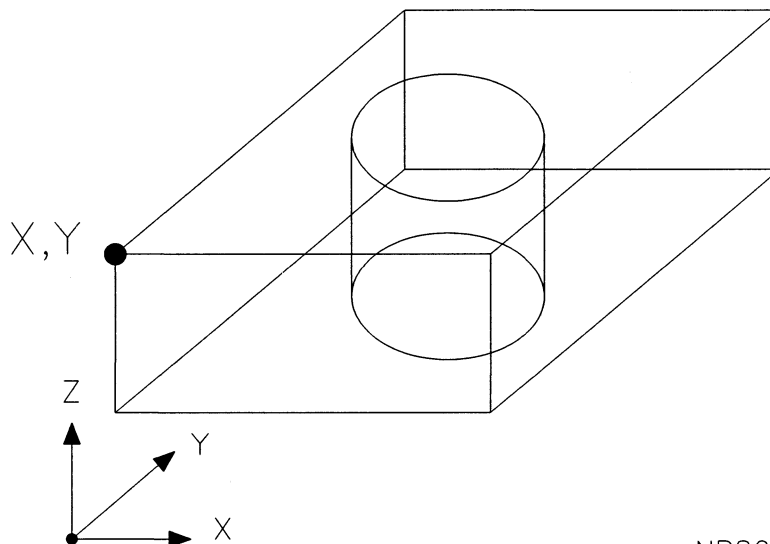
Explanation:

- N1: Set the program zero point
- N2: Set the plane of operation to be the XY-plane
- N3: Set the graphic window to define the 3D space
- N4: Start of contour description section.
- N5: Define start point of outer contour.
- N6-N13: Describe the outer contour. The coordinates are related to the point programmed in the G199-block. The geometry of the control is used to program the contour.
- N14: End of contour description section
- N15: Load the tool and set the spindle speed
- N16: Start the spindle and move the tool to the start point.
- N17: Move tool to depth
- N18: Move tool to the contour
- N19: Set radius compensation RIGHT
- N20: Move along the contour. With the G14 function the programmed contour in the outer contour description is used.
- N21: Cancel radius compensation
- N22: Move tool to the start point and away in the tool axis. End of program

69. Begin contour description G199

Purpose

1. To define the position of the workpiece blank contour related to the program zero point or machine zero point. This position is used during the graphic simulation of the program run.
2. To define the position of any machine part with which the tool might collide. Collision to be detected during the graphic simulation.
3. Drawing a contour during the wireplot simulation.



NB8679a

Format

To define a blank contour

N... G199 [Coordinates of position] B1 {C1} {C2}

To define a machine part

N... G199 [Coordinates of position] B2 {C1} {C2}

To draw a contour during the wireplot simulation.

N... G199 [Coordinates of position] B3 {C1} {C2}

Parameters

X,Y,Z Startpoint coordinate

B Model 1=mat.,2=mach.,3=contour

C Zero point 1=Machine,2=Workpiece

Associated functions

G98, G99, G195 to G198

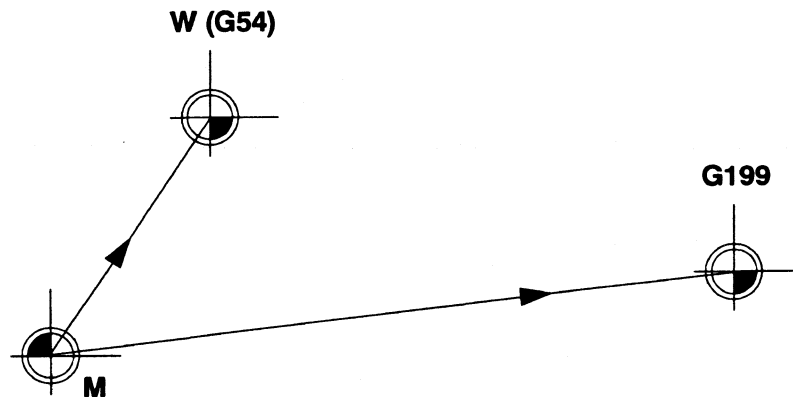
Type of function

Non-modal.

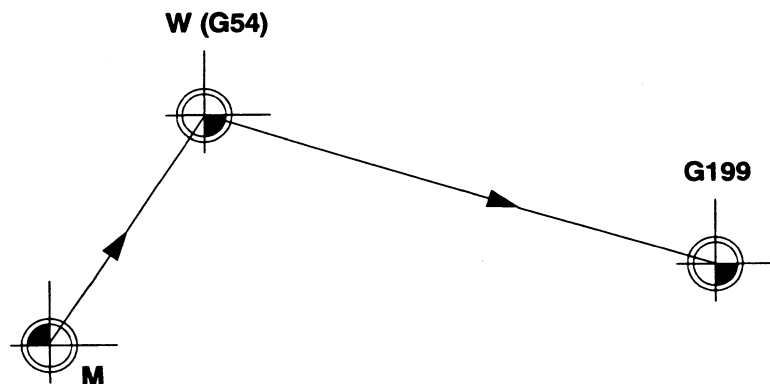
Notes and usage

ZERO POINT (C1/C2)

The position of the contour can be related to either the machine zero point M_0 or the program zero point W.



C1 the contour position is related to the machine zero point M_0 .



C2 the contour position is related to the program zero point W.

DEFAULT ZERO POINT

If the C-word is not programmed the following default settings are used:

For a workpiece blank the G199 coordinates are related to the program zero point W (C2).

For machine parts the G199 coordinates are related to the machine zero point M_0 (C1).

COORDINATES OF CONTOUR POSITION

The coordinates of the contour position must be absolute, cartesian coordinates. Polar coordinates are not allowed. These coordinates are related to the point defined with the C-word

Defining the position with E-parameters is not possible. Also defined points cannot be used.

TYPE OF CONTOUR DESCRIPTION (B)

A contour description can be used for three purposes:

- B1: to define a blank contour (uncut part)
- B2: to define machine parts, such as clamping devices, etc.
- B3: to draw a contour of a machined work piece.

WORKPIECE BLANK (B1)

The outer and inner contour of a workpiece blank, thus the form of the uncut material, can be described with the graphic functions. The cutting of this material to the required shape can be simulated on the control.

Refer to the functions G197/G198 for details about describing a contour.

MACHINE PARTS (B2)

The contours of machine parts, such as clamping devices, can also be described by using the G199 function. This allows possible collisions between the cutting tool and machine parts to be detected.

The details given for the description of workpiece contours also apply to machine parts, except the B-word in the G199 block which must be B2.

The description of machine parts is usually at the beginning of the partprogram. However, the description blocks can also be placed later in the program, in a particular part of the machining cycle.

A number of shapes can be programmed in succession enabling a machine part to be made up of several layers.

Several machine parts can be defined in a part program.

With macros a library for machine parts can be built up.

CONTOURS (B3)

The contours of a machining part (outside contour rectangular pocket) can be describes with the graphical function.

The contour is only visible in wire plot simulation.

For special information about the contour description see the functions G197/G198.

The special information of B1 and B2 is also valid for the B3.

The contours can be describes all over in the program. More contours are possible.

There is no differences between G198 and G197 contours.

The contours are be drafted in the colour Cyan.

TOOL IMAGE

A tool image can be assigned to the tool with the aid of the G-word in the tool memory. The required image can be selected from a set of available tool images and is used by the CNC system to accurately simulate the machining.

Refer to the user manual for selecting the tool image.

GRAPHIC FUNCTIONS AND MACROS

The function for starting the contour description (G199), the contour description itself (G198 and / or G197) and the function for ending it (G196) must be in the same partprogram or subprogram (macro).

E-parameter can be used with the function G199, but parameter definition is not allowed in the G199 block itself.

ANOTHER BLANK CONTOUR DESCRIPTION

In a part program more than one contour for a blank workpiece can be described. Each contour must be defined with the functions G199, G198, G197 and G196. A new graphic window (G195) must be defined too.

Only one blank contour is shown on the display. As soon as another description is encountered, the previous one is deleted and the other model shown. The tool movements are shown in the displayed model.

PLANE SELECTION (G17,G18,G19)

If a function for plane selection (G17, G18, G19) is encountered, the displayed program is deleted. After the G-function for plane selection another contour description can be given. The programmed movements are displayed in this model.

Only the movements in the last programmed plane are shown.

CYLINDER INTERPOLATION

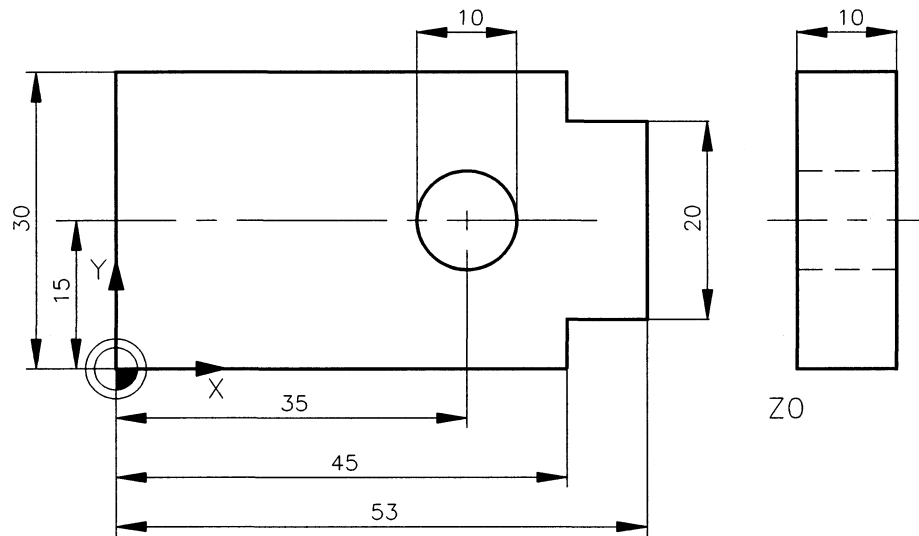
Movements programmed in the plane of the cylinder (G182) cannot be visualized on the display of the control.

RESTRICTIONS

1. A G198 function must immediately follow the G199 block.
2. The G199 function cannot be used in MDI or TEACH-IN (PLAYBACK) mode.
3. A contour description should not be repeated with a G14 at another place. In this case the control repeats the model and the operations executed in this model before the G14 and does not stop properly. It is advised to write the blocks again in the program or to put the contour description in a macro.
4. A contour description is only allowed in the basic coordinate system, thus G180 active.

OPERATION WITH A CONTOUR DESCRIPTION

1. A described contour between a G199 and a G196 is treated as one block. This means that at SINGLE BLOCK the complete contour is executed in one step.
2. A MANUAL BLOCK SEARCH to a block between a G199 and a G196 is not possible and results in an error message.
3. INTERVENTION during the drawing of the contour is not possible. The INTERVENTION is executed once the complete contour is drawn.

Example**EXAMPLE 1.**Defining a blank contour with clamping devices

NB9809

Each clamp is described in a separate macro. With two parameters the start point of the clamp contour is programmed:

E1: X-coordinate of the contour start point related to the program zero point.

E2: Y-coordinate of the contour start point related to the program zero point.

The contour itself is programmed with fixed dimensions.

If this clamp is used in different part programs, the two parameters must be set at the macro call and then the clamp can be used for graphical purposes.

Macro for the left clamp

```

N1991
N1 G92 X=E1 Y=E2
N2 G199 X0 Y0 Z0 B2 C2
N3 G198 X0 Y0 Z0 D10
N4 G1 X45
N5 Y5
N6 X53
N7 Y25
N8 X45
N9 Y30
N10 X0
N11 Y0
N12 G197 X30 Y15 D-10
N13 G2 I35 J15
N14 G196
N15 G92 X=-E1 Y=-E2

```

Macro for the right clamp

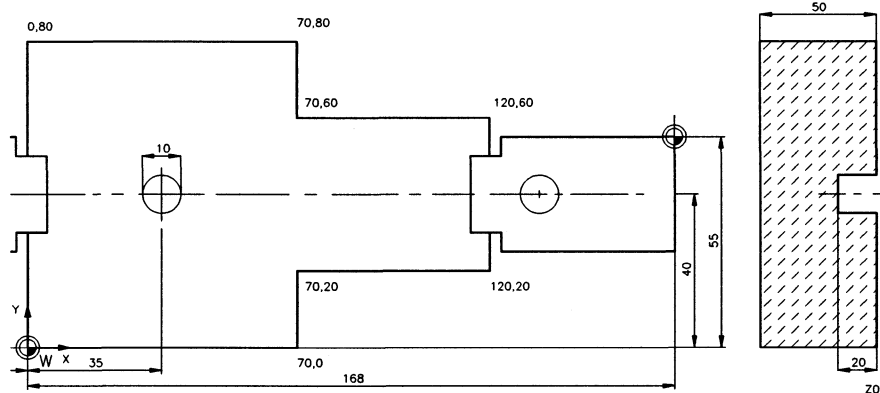
```

N1992
N1 G92 X=E1 Y= E2
N2 G199 X0 Y0 Z0 B2 C2
N3 G198 X0 Y0 Z0 D10
N4 G1 X-45
N5 Y-5
N6 X-48
N7 Y-25
N8 X-45
N9 Y-30
N10 X0
N11 Y0
N12 G197 X-30 Y-15 D-10
N13 G2 I-35 J-15
N14 G196
N15 G92 X=-E1 Y=-E2
    
```

Explanation of the macros

- N1: A zero point shift to let the start point of the clamp coincide with the program zero point.
- N2: Start the contour description section of the clamp. Its zero point is indicated in the drawing. Coordinates related to the program zero point are used.
- N3: Start of the outer contour description of the clamp. Offset values relative to point stated in G199-block.
- N4/N11: Describe the outer contour. Coordinates are related to the point defined in the G199- block. The depth D is measured from the surface (Z0 in G199).
- N12: Start of the inner contour description. Offset values relative to point stated in G199- block. The depth D is measured from the surface (Z0 in G199).
- N13: Describe the inner contour to be a hole. Centre point coordinates are related to the point defined in G199-block.
- N14: End of the description section of the clamp.
- N15: Restore the program zero point.

EXAMPLE 2. The graphic section of the part program:



NB9810

N199000
 N1 G17
 N2 G54
 N3 S1200 T1 M6
 N4 G195 X-20 Y-20 Z-60 I180 J110 K70
 N5 G199 X0 Y0 Z0 81 C2
 N6 G198 X0 Y0 D-50
 N7 G1 X70
 N8 Y20
 N9 X120
 N10 Y60
 N11 X70
 N12 Y80
 N13 X0
 N14 Y0
 N15 G197 X30 Y40 D-20
 N16 G2 135 J40
 N17 G196
 N18 G22 N=1991 E1=48 E2=25
 N19 G22 N=1 992 E1 =1 68 E2=55

N200 M30

Explanation:

N1:	Set the main plane to be the XY-plane
N2:	Set the zero point
N3:	Load the tool and set the spindle speed
N4:	Set the graphic window to define the 3D space
N5:	Start the contour description section of the workpiece blank. The start point of the contour coincides with the program zero point.
N6:	Start of the outer contour description. Offset values relative to point stated in G199-block.
N7/N14:	Describe the outer contour. Coordinates are related to the point defined in the G199-block. The depth D is measured from the surface (Z0 in G199).
N15:	Start of the inner contour description. Offset values relative to point stated in G199-block. The depth D is measured from the surface (Z0 in G199).
N16:	Describe the inner contour to be a hole. Centre point coordinates are related to the point defined in G199-block.
N17:	End of the description section of the blank contour.
N18:	Call the macro for the left clamp
N19:	Call the macro for the right clamp

Begin contour description G199

70. Create pocket cycle macro's G200

Purpose

The Universal Pocket Cycle allows easier and quicker production of CNC programs which control the milling of rectangular and circular pockets. 'Islands' of uncut material within the pockets can also be produced.

Format

N... G200

Associated functions

G200...G208

Type of function

Non

70.1 Introduction universal pocket cycle

The Universal Pocket Cycle (UPC) allows easier and quicker production of CNC programs which control the milling of rectangular and circular pockets. 'Islands' of uncut material within the pockets can also be produced. The CNC can calculate the minimum number of starting points to produce the required pocket contours in the shortest time. Tool movements are based on using the contour-parallel method which is the most effective technique for machining pockets.

A 'start point' macro can be generated and used to pre-drill the cutter start position(s) in the work piece. The minimum number of start points required to machine the pocket are calculated, keeping machining times to a minimum.

The programmer only needs to state the program numbers of the required UPC macros (subprograms), together with the machining parameters such as the feed rate, clearance distance, cutter radius etc., and the start point and dimensions of the pocket contour.

The CNC uses the above data to calculate the start points and coordinates of the tool paths which will be parallel to the contour's sides. These calculations are performed before the pockets are milled. A UPC macro is calculated only once; the contents of this type of macro will be re-used whenever the macro is reactivated in the CNC program.

The macro is calculated only once. The macro contents are reused every time the macro is called in the CNC program. If changes have been made (e.g. different cutter), the macro is recalculated.

The tool movements proceed to a contour-parallel process, which enables milling pockets to be machined most economically.

A 'finishing' cutting path can be incorporated in a machining cycle as a separate UPC macro.

This function must be programmed before the universal pocket cycles that have to be calculated, and commands that:

- if the calculations have not yet been made, the coordinates of the cutter paths must be calculated
- the cutter paths will be included in a macro generated by the CNC; the number (N1=...) of this machining macro is programmed in the G201-block
- if necessary (stated in a G201-block, by N2=...), a second macro will be generated for drilling the starting points
- if necessary (stated by the G203- or G205-function), the macros (N1=...) for finishing the contours will be generated.

All operational conditions such as the machining plane, zero point shifts and tool offsets, must be active before the G200-function is executed.

All universal pocket cycles which have been programmed between a G200- block and G202 or M30 will be calculated (if necessary).

A G200-block may be included in a macro, however the pocket will only be searched for in macros which are nested deeper.

The CNC calculates the UPC macros before executing the CNC partprogram, therefore, any blocks between G200 and G201 will initially be ignored. After the macros have been generated, these blocks will then be executed.

For reasons of memory space the pocket contour and associated islands may not exceed 50 sides. 15KB of additional memory capacity will be necessary to allow approximately 50 sides to be stored. A circular movement that is greater than, or equal to, 180° will be treated as if it had two sides because the CNC system automatically divides this type of movement into two equal parts.

The machining plane (G17/G18/G19) must be selected before the G200-function or after a G202-function, is executed.

If the coordinates of the defined points are changed after the pocket has been calculated, and if the pocket must be calculated again, the macros generated during the calculation of the second pocket will be stored. The macros of the first calculation will be destroyed.

For calculating the macros, characteristics and quantities such as programmed points, scaling, rotations, mirror image etc. will be used the way they are active with the G200-block concerned.

If errors are generated in a pocket cycle program or graphics shows material left, the programmer is advised to make alterations to the program in order to prevent those errors. Proposals are changing the overlap percentage, change programming sequence, change entry point, split up program, define pseudo-island of 1 micron.

If a calculation error occurs during execution of a pocket cycle (0170, 0176) please check the pocket contour. If the contour seems ok check the toolradius size with respect to contour and islands, and choose a tool with a smaller radius when the originally programmed tool is comparatively large.

If in pocketcycles the toolradius is relatively large compared to the pocket or its islands, some problems may occur:

- G201 i... Certain i-values may lead to material left or damage of islands programmed.
- Errors P163, 0170 or 0176 could be set.

The only cure in above cases is to chose a smaller tool for machining these pocket cycli concerned.

70.2 Partprogram structure

The example below shows a simple program which uses a macro which specifies a pocket cycle.

```

N99999 G54
N1 G17
N2 \
:      > normal machining
N96 /
N97 G200          Begin pocket cycle
N98 G81           Drill cycle description
N99 G22 N=..      Start point drilling
N100 G201 N1=.. N2=.. Starting to mill de pocket cycle
N101 G203 N1=..      Begin pocket contour description
N102 \
:      > pocket contour description
N109 /
or
N101 G208 N1=..      pocket contour description of a regular quadrangle.
N110 G204           end pocket contour description
N111 G205 N1=..      start island contour description
N112 \
:      > island 1 contour description
N118 /
N119 G206          end island contour description
N120 G205 N1=..      start island contour description
N121 \
:      > island 2 contour description
N129 /
N130 G206           end island contour description
N131 G207 N=..       island contour description 3 is a macro
N132 G202           end of pocket cycle

N350 G22 N=..      pocket contour finishing
N351 G22 N=..      island 1 finishing
N352 G22 N=..      island 2 finishing
N353 G22 N=..      island 3 finishing

```

70.3 Translation, rotation and mirror image of a pocket

A pocket cycle can be described using a datum point which is different from the datum point used during program execution; this may be necessary for either arithmetical reasons, or if the pocket is included in a macro.

A different datum point is established by programming a datum point shift and/or axis rotation before the G201-block.

The datum point shift and/or axis rotation are programmed with the standard function G92 or G93. The axis transformation will then be performed on the generated machining macro. When the pocket has been cut the programmer must ensure that the program datum point is reset at the correct location. Similarly, when using the macro for the starting points and macros for finishing the pocket contour, the programmer must ensure that the correct datum point is used.

A pocket may be used again within the same part program by programming a new start position and orientation.

Example:

```
%PM
N9001
:
:
:
N90 G200
N100 G201 N1=9999
N110 G203
:           \
:           >      description of the pocket contour
:           /
N200 G204

N205
:           \
:           >      description of an island contour
:           /
N206

N300 G202
:
:
:
N400 G92 X.. Y.. Z.. B4=..
N410 G22 N=9999
```

In block N400 the datum point is shifted and the pocket rotated (B4=..), so that the starting point of the pocket is correctly positioned. In block N410 the machining macro is called. The pocket will be cut again but at a different location.

In this example (with G17 active), mirroring about the Y-axis is programmed in block N300 and the machining macro executed in block N310 as mirror image. The disadvantage is, that one pocket is machined in a backward direction and the other in a forward direction. When this is not possible for technological reasons, the mirror image function cannot be used. The mirrored pocket must then be programmed again.

70.4 Same pocket in another program

If a pocket (and associated islands) occurs in different programs, the complete pocket may be written in a macro (subprogram); this macro will then be called at that point in the partprogram where the pocket must be cut.

The functions G201, G202, G203/G204, and G205/G206 have to be stated in the macro.

Programming will be:

```
%MM
N9001 G201 Y.. Z.. B.. N1= N2=
N1 G203
:           \
:           >   pocket contour
:           /
N8 G204

N9 G205
:           \
:           >   contour of island 1
:           /
N13 G206
N14 G205
:           \
:           >   contour of island 2
:           /
N18 G206
N19 G202
```

Represents a macro of a pocket which has islands.
A partprogram which uses this macro could look like:

```
%PM
N9999
N1 G200
:
N50 G22 N=9001
:
```

A datum point shift prior to the macro call positions the pocket in the correct location.

Remark: By including the G201-block in a macro, the nesting level of the machining macro will be increased by one. Macros cannot be nested more than 8 times.

70.5 Operating section

Pocket, starting points and finishing macros

Generation

When the control encounters a G200-block in a program, it searches for the pocket functions (G201-block + associated contour description + G202-block). Upon finding a G201-block (in the part program or possibly via a macro call) and if no macro carrying the programmed macro number (N1=.. in the G201-block) exists, a pocket macro will be generated by the control. (The data which determine the pocket macro, viz. the parameters Y, Z, B, R, I, K in the G201-block and the contour description, will be stored with the pocket macro).

If a macro with the programmed pocket macro number exists already, the associated data will be compared with the programmed data. If they do not match, a new pocket macro will be generated.

A new starting points macro is generated if:

a starting points macro number is programmed (N2=.. in the G201-block) and a new pocket macro is generated or if no macro with the programmed starting points macro number exists.

Finishing macros are generated if:

a finishing macro number is programmed (N1=.. in the G203- or G205-block concerned) and a new pocket macro is generated or if no macro with the programmed finishing macro number exists.

REMOVAL

After being generated the macros are "locked". In order for them to be removed they must first be "unlocked". Removal may be necessary, for example, if a G201-block is being removed from a program or if an 'N1=' or 'N2=' address in a G201-block or an 'N1=' address in a G210-block is being changed.

The macros will be automatically removed if a program is started in which the pocket definition has been changed.

In V330 are for the operator, the generated macros in the macro memory no longer visible. For using a macro in an other program, the macro number must be given in the macro memory. After that the macro will be visible in the macro memory. It is possible to read-in and read-out the macro program.

SIGN OF LIFE

The Universal pocket cycle software is designed to stop all calculations upon detection of an error. To indicate that the CNC is busy a rotating indicator, with the word "CLOCK", is displayed on the screen. While the indicator is rotating the CNC is busy calculating the pocket cycle.

TEACH-IN

The use of a pocket G-function (G200 to G208) in TEACH-IN MDI and TEACH-IN/PLAYBACK is not permitted.

EDITING

During editing, static or dynamic programming support may be used.

A (new) program may be edited while a program with a pocket section is running.

BLOCK DELETE

If a G201-block contains a "/" character and "block delete" is activated, then:

G200 will be executed in the normal way, i.e. macros will be generated if necessary. the G201-block, the contour description and the G202-block will be skipped during the program execution, i.e. program execution will continue after the G202-block.

BLOCK SEARCH

Searching a G200-block causes pocket macros to be generated if they are not yet present.

During program execution (after the G200-block has been executed), G201 is taken as a macro call (G22) to the pocket macro. However, after executing the pocket, a jump is done to the first block after the terminating G202-block.

Searching for a block in a pocket macro is performed in the same way as searching for a block in a macro called via G22.

Note: As G201 is taken as a macro call during program execution, the nesting level (max. 8 levels) should be taken into account when programming G201.

INTERVENTION

Intervention during the execution of a G200-block is possible. However, macros have to be recalculated after an intervention and this could take a long time if a complex contour is required.

Execution can be resumed via "start". After intervention, it is not possible to enter the edit mode without a "clear control".

Intervention during the execution of a pocket macro is dealt with in the same way as intervention in a macro called with G22.

INCOMPLETE PROGRAMMING

Normally, the following G-functions are programmed for a pocket cycle: G200, G201, G202, G203/G204, G205/G206, G207 and if necessary, G208.

G201 and G202, G203/G204 and G205/G206, are required to appear as a combination in the same program or the same macro, otherwise an error message is issued.

If a G203/G204 or G205/G206 combination appears (possibly via a macro call G22) without the associated G201/G202 combination, an error message is issued.

A G207 should always appear after a G205/G206 combination, with both G205/G206 and G207 belonging to the same G201-block, otherwise an error message is issued.

If a G201/G202 combination is programmed without contour descriptions, an error message is given.

If G200 is programmed without G201/G202 contour descriptions, the program is executed in the normal way (without pocket cycle).

If G201/G202 contour descriptions have been programmed without G200, no macros are generated and G201 is taken as G22,

OPERATION MODE CHANGE

An operational mode change can only occur after a G200-block has been completely executed.

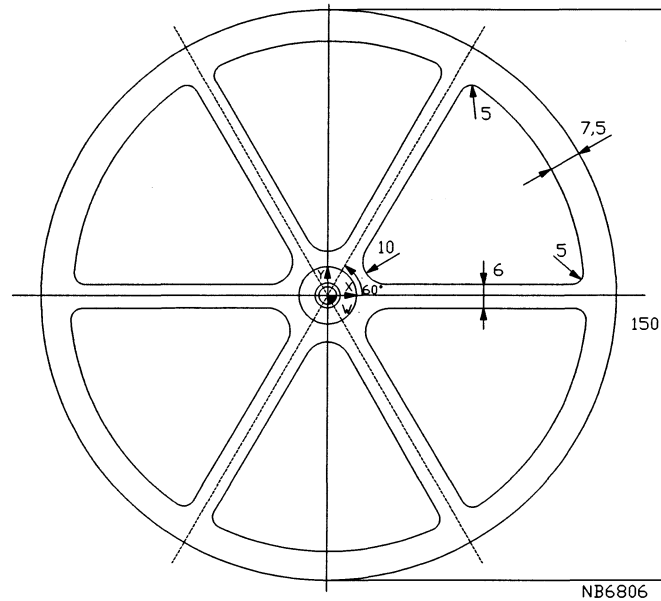
70.6 Error messages

Number	Message
#5 O70* Solution:	Memory pool exhausted. Pocket-cycle calculation error (Operating error). Decrease number of parameters (MC 83).
O176*	Module xxx, number xxx.
Note:	The 0170 and 0176 messages are displayed together. Solving this problem can be achieved by using one of three methods. <ul style="list-style-type: none"> - change the tool radius. - change the overlap I-word value. - change the amount of stock to be left for the finishing cycle.
P07	Programmed data is out of range. In a block with a circular (G2/G3) side programmed with a radius value. When the programmed radius is two times greater than the distance between the start and end point. This can occur on a side of 180°. The error is removed by changing the start or end point.
P35	End point not on circle. On a block with a circular (G2/G3) side programmed with centre point or a block with a circular (G2/G3) side programmed with geometry. The value of the programmed centre point is greater than the value of the Circular end point window (calculated centre point between start and end point) specified by machine constant 712. The error can be removed by changing the centre point or the start/end point.
P75	Circle without centre point. An are has been programmed which has the same start and end point. The error is removed by deleting the program block or by programming an are which has a separate start and end point.
P140	Invalid G207 nesting.
P141	Too many sides programmed.
P142	Too many contours programmed.
P143	Invalid 'G' in pocket-cycle mode.

P144	Invalid contour description. This error is caused by one of the following reasons: - the G208-function has been omitted - the G208-function specifies roundings (radii) which are too large - the G208-function block contains X=0 or Y=0 - the G208-function block contains B1 > 180° - the G208-function block contains an I or R-word.
P145	A finishing cutting path cannot be generated from the current description, another start point must be chosen .
P146	No G202 defined.
P147 P148	Memory manager error, Floating point error.
P150	Tool not found.
P160	Pocket macro generation error.
P161	Finishing macro generation error.
P162	Macro start point generation error.
Note	Errors P160, P161 and P162 are produced by: - a full CNC memory - too many start points having to be calculated.
P170	Contour xx not closed.
P171	Contour xx has more inner areas.
P172	Contour xx intersects contour xx.
P173	Contour xx enclosed by contour xx.
P174	Contour xx is outside the pocket.

70.7 Example

EXAMPLE 1. Rotated pocket



```

N3620511 (WHEEL AS POCKET)
N1 G17
N2 G54
N3 G195 X-90 Y-90 Z0 I180 J180 K-10
N4 G99 X-85 Y-85 Z0 I170 J170 K-10
N5 G200
N6 T31 M6 (DRILL RADIUS 4. mm)
N7 G81 Y1 Z-5 F100 S100 M3
N8 G22 N=3620501
N9 G92 B4=60
N10 G14 J5 N1=6 N2=7
N11 G93 B4=0
N12 T04 M6 (ROUGHING MILL RADIUS 3. mm)
N13 S1500 M3
N14 G201 Y0.1 Z-5 B1 150 R3 F1000 N1=3620500 N2=3620501 F2=500
N15 G203 X37.5 Y3 Z0 N1=3620502
N16 G64
N17 G1 X1=0 Y1=3 B1=0 J1=2
N18 G3 R5
N19 I0 J0 R67.5 J1=1
N20 R5
N21 G1 X1=0 Y1=0 B1=-120 I1=-3
N22 G3 R10
N23 G1 X37.5 Y3 B1 =0
N24 G63
N25 G204
N26 G 202
N27 G92 B4=60
N28 G14 J5 N1=11 N2=24
N29 G93 B4=0
N30 T3 M6 (FINISHING MILL RADIUS 2.5 mm)
N31 S1800 M3
N32 G22 N=3620502
  
```

```
N33 G92 B4=60  
N34 G14 J5 N1=28 N2=29  
N35 G0 Z100 M30
```

Explanation:

The CNC processes the blocks N3 and N4 which define a graphic simulation of the program's operation.

The UPC macros are calculated first (blocks N5 and N14 to N26 are executed). The following macros are created:

- Macro No. 3620500 for the machining-cycle;
- Macro No. 3620501 for the starting-points;
- Macro No. 3620502 for the finishing-cycle.

Tool 31 is selected and a drilling cycle defined by the function G81. CNC control is transferred to UPC macro no. 3620501 by the use of E-parameter no.1 and the G22-function. The first starting points are drilled. Block N9 commands the macro's coordinates to be rotated 60° in a counter clockwise direction.

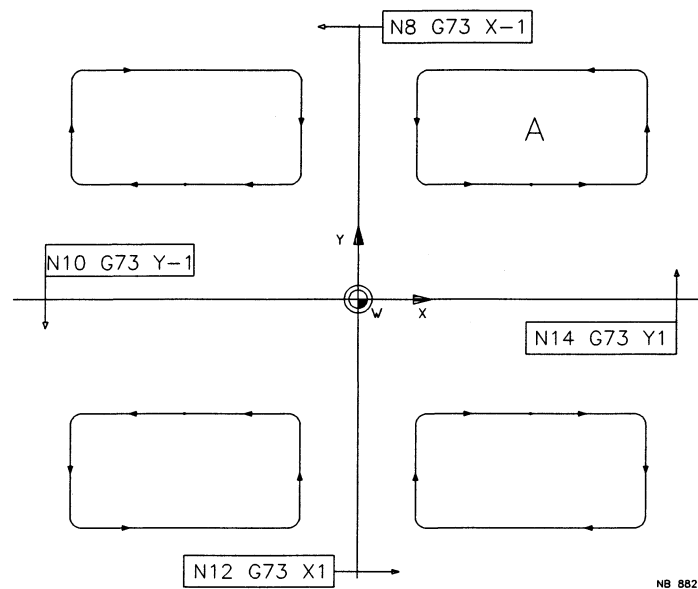
Tool 4 is selected. Block N14 commands that the machining cycle (macro no. 3620500) is executed. When the machining cycle is finished, the coordinates of the macro 3620500 are rotated

60° in a counter-clockwise direction. Block N28 commands the machining cycle to be repeated and rotated five times (Note: N2=27 can be omitted from the block and the repeat instruction will still operate).

Tool 3 is selected. Block N32 assigns the value 3620502 to E-parameter no.1 . The 'finishing' cycle is executed in the same manner as for the two previous UPC macros.

Block N35 commands the tool to move to Z100 and the partprogram to end.

EXAMPLE 2. Mirrored pocket



NB 8827

A: Programmed pocket cycle

N3620513 (MIRROR IMAGE OF A POCKET)

N1 G17

N2 G54

N3 G98 X-150 Y-110 Z0 1300 J220 K-10

N4 G99 X-145 Y-105 Z0 1290 J210 K-10

N5 G200

N6 T31 M67 (DRILL RADIUS 4. mm)

N7 G81 Y1 Z-5 F500 S1000 M3

N8 E1=3620501

N9 G22 N=E1

N10 G73 X-1

N11 G14 N1=7

N12 G73 Y-1

N13 G14 N1=7

N14 G73 X1

N15 G14 N1=7

N16 G73 Y1

N17 T4 M6 (ROUGHING MILL RADIUS 3. mm)

N18 S1800 M3

N19 G201 Y0.1 Z-5 B1 160 R4 N1 =3620500 N2=3620501 F1000 F2=500

N20 G203 X75 Y50 Z0 N1 =3620502

N21 G1 X120

N22 G3 X125 Y55 R5

N23 G1 Y95

N24 G3 X120 Y100 R5

N25 G1 X30

N26 G3 X25 Y95 R5

N27 G1 Y55

N28 G3 X30 Y50 R5

N29 G1 X75 Y50

N30 G204

N31 G202

```
N32 E1=3620500
N33 G14 N1=8 N2=14
N34 T3 M6 (FINISHING MILL RADIUS 2.5 mm)
N35 S2000 M3
N36 E1=3620502
N37 G14 N1=7 N2=14
N38 G0 Z100 M30
```

Explanation:

The CNC first processes blocks N3 and N4 which defined a graphics simulation of the partprograms's operation.

The UPC macros are calculated first (blocks N5 and N19... N31 are executed). The following macros are created:

- Macro No. 3620500 for the machining-cycle;
- Macro No. 3620501 for the starting-points;
- Macro No. 3620502 for the finishing-cycle.

Tool 31 is selected and a drilling cycle defined by the function G81. CNC control is transferred to UPC macro no. 3620501 by the use of E-parameter no.1 and the G22-function. The first starting points are drilled.

Blocks N10 to N16 command the coordinates of the starting-points macro to be mirrored about the X and Y axes and for the macro to be executed after each mirroring. The G22-function transfers CNC control to the macro and the G14-function commands the block N9 to be repeated once.

Tool 4 is selected after the pocket has been executed four times. Block N32 assigns the value of 3620500 to E-parameter no.1; block N33 commands that blocks N10 to N16 are repeated. The 'machining' macro is therefore executed once and then mirrored and repeated in the same manner as for the 'starting-points' macro.

Tool 3 is selected. Block N36 assigns the value 3620502 to E-parameter no.1 . The 'finishing' cycle is executed in the same manner as for the two previous UPC macros.

Block N38 commands the tool to move to Z100 and the partprogram to end.

71. Start contour pocket cycle G201

Purpose

Entering technological data to calculate the pocket cycle. Pocket milling starts with this block.

Format

N... G201 N1=.. Z.. {N2=..} {Y..} {B..} {R..} {I..} {J..} {K..} {F..} {F2..}

Parameters

Y Stock removal
Z Total pocket depth

B Clearance
R Tool radius for calculation
I Cutting width mill in %

J J1:climb / J-1:conventional
K Cutting depth
F Feed pocket milling
F2= In depth feed
N1= Milling macro number
N2= Startpoint macro number

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

Type of function

Non

Notes and Usage

N1=: Number of the machining macro. This number has to be programmed.

N2=: Number of the starting points macro.

Y: Machining allowance = amount of material required to be left on the contour for finishing The Y-word carries no sign. If Y has not been programmed, Y=0 will be used as a default.

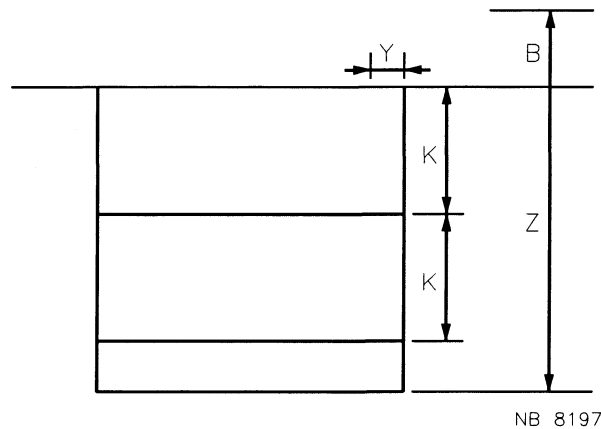
Z: The total depth of the pocket. The depth is measured from the coordinate of the tool axis from the G203-block (which is equal to the position of the upper surface of the pocket). A negative Z-word value is the depth in the negative direction of the tool axis. A positive Z-word value is the depth in the positive direction of the tool axis. The Z-word must always be present in a block which contains the G201-function.

These words are independent of the selected machining plane.

- B:** Clearance distance above the pocket. This distance is measured from the coordinate of the tool axis specified with the G203-function.

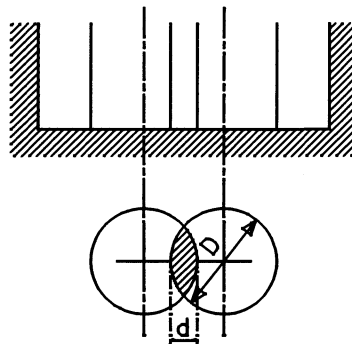
positive B-word value is the clearance distance in the positive direction of the tool axis; a negative value is the clearance distance in the negative direction. The sign of the B-word must always be opposite to the sign of the Z-word. If a B-word is not been programmed, B=0 will be used as a default.

The B-word is used to define the position on the tool axis where the feed movement begins for entering the pocket at each starting point. At the end of the machiningoperation the tool is retracted above the pocket surface, at a distance stated by the B- Word.



- R:** The cutter radius to be used for calculating the cutter paths. The final cutter radius which is actually used can be different. The R-word carries no sign and must always be programmed.
- I:** The amount of overlap between cutting passes. The overlap distance is specified as a percentage of the tool's diameter eg. I75= 75% of the diameter.

The I-word carries no sign. If the I-word is not programmed, the value of MC720 will be taken.



NB 0617

d= overlap distance
D= Cylinder cutter diameter
 $I = 100 (1 - (d/D))$

- J: The direction of movement for rough milling.
J1 (default value): counter-clockwise direction. J-1: clockwise direction.
- K: The cutting depth per path. The K-word carries no sign. The last feed-in distance may be smaller than the value of K for the final cutting pass. If the value of K is larger than that of Z, or K-word is not stated, the Z-value will be used ie. the value of $K = |Z|$.
- F: The feed during milling.
If F is not programmed, the last programmed feed will be used.
- F2=: The feed for moving to a next machining plane. If the holes have been pre-drilled a great value can be taken. If F2= is not programmed, the last programmed feed will be taken.

The functions G90, G40 and G63 will be automatically activated when the G201-function is executed. G90 is required to be active because the generated macros use absolute dimensions and the first position of the pocket contour definition (G203-block) must be absolute.

71.1 Usage of the generated macros

71.1.1 Starting points macro

The control is capable of generating a starting points macro. The starting points macro contains the points where the cutting tool enters the material for "roughing out" the pocket (these points are calculated by the control). The programmer can command a hole to be drilled in these positions, so that the cutting tool need not cut in the direction of the tool axis.

The macro generated by the CNC will be similar to that given below:

N (= N2-word of the G201-block)
 N1 G90 Absolute programming
 N2 G79 X.. Y.. Z.. Activate a predefined drilling cycle
 N3 G79 X.. Y.. Z..
 N4 G79 X.. Y.. Z..

The position in the machining plane XY (G17), XZ (G18) or YZ (G19) is given in the coordinates of the axes system in which the pocket has been described. The tool axis Z (G17), Y (G18) or X (G19) is given by the G203-block of the pocket contour.

After execution of the starting-points macro, G90 becomes active. The starting-points macro is generated simultaneously with the machining macro.

A partprogram could be as follows:

```
%PM
N9900
N1 G200
:
N90 T1 M6
N100 G81 X.. Y.. Z.. B.. F.. S.. M..
:
N110 G22    N=9902
:
N200 G201 Z..      N1=9901      N2=9902
:
:                                \
:                                > description of the pocket (including islands)
:                                /
N300 G202
```

:

The cycle for pre-drilling the starting points is defined in block N100. The starting points macro is called in block N110. The cycle from block N100 is executed on the starting points.

71.1.2 Machining macro

The machining macro is generated by the control and includes all the movements necessary for roughing out a pocket. This macro (subprogram) will be called when the G201-function is executed. When the macro is terminated by the G202-function, the G40 and G90-functions will always become active automatically.

The example below shows a simple program which uses a macro which specifies a pocket cycle.

```
% N9090
N1    G40                                (No radius correction)
N2    G90                                (Absolute coordinates are used)
N3    G0      X...      Y...      Z...    (Tool moves to start point)
N4    G91                                (Incremental coordinates are used)
N5    G1      Z...      F...              (Tool is fed to depth)
N6    G90                                (Absolute coordinates are used)
N7    F...                                (Feedrate)
:      \
:      >      (Pocket milled)
:      /
N99
N100  G91
N101  G0      Z...                        (Tool is retracted)
N102  G90                                (Absolute coordinates used again)
N103  G0      X...      Y...              (Tool moves back to starting point)
N104  G14     N1=4      N2=103      J...   (Pocket milling is repeated)
N105  G91                                (Incremental coordinates are used)
N106  G1      Z...      F...
N107  G14     N1=6      N2=99            (Final cutting pass is performed)
N108  G0      Z...                        (Tool retracts out of workpiece)
```

71.1.3 Macro for finishing a pocket contour

The macro for finishing the pocket contour is generated by the control. The macro includes all the movements including a circular or linear feed-in and feed-out, necessary for finishing the pocket sides.

71.1.3.1 Feed-in point

The feed-in point S is determined by the CNC. This point is always near the start position of the pocket and at an equivalent distance (d) to both first and last sides.

The CNC will calculate different feed-in points depending on which contour side the tool has to first move to. The programmer therefore has to determine the correct tool size and the order in which the contour sides are stated, to prevent the tool colliding with the workpiece.

The CNC uses the formula below to calculate point S.

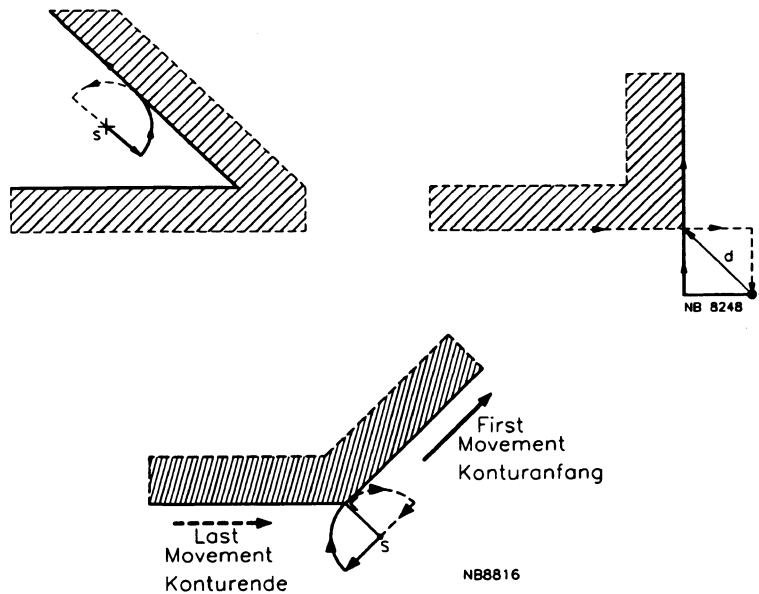
$$\text{distance} = \text{factor} * \text{radius} : 100$$

where:

factor: given per machine constant (MC 719) (between 101 ..200).
radius: the value of the R-word from the G201-block

Within the circle, with S as centre and distance as radius no contour element of an island or pocket is allowed.

If it is not possible to calculate point S, the CNC generates an error code.

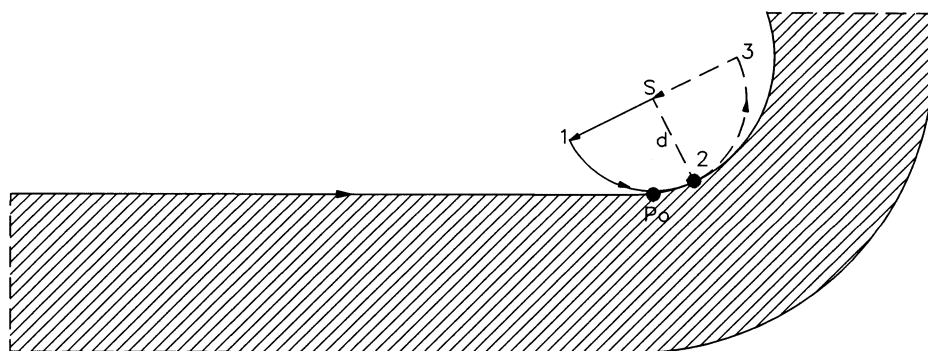


The feed-in movement is linear when the first and last movements are also linear and the angle between \Rightarrow 270 degrees (this angle is measured inside the pocket). A circular feed-in movement is used in all other instances.

If a different tool is used for the finishing cut the programmer must ensure that the tool does not collide with the contour.

71.1.3.2 General format of the macro for a circular feed-in

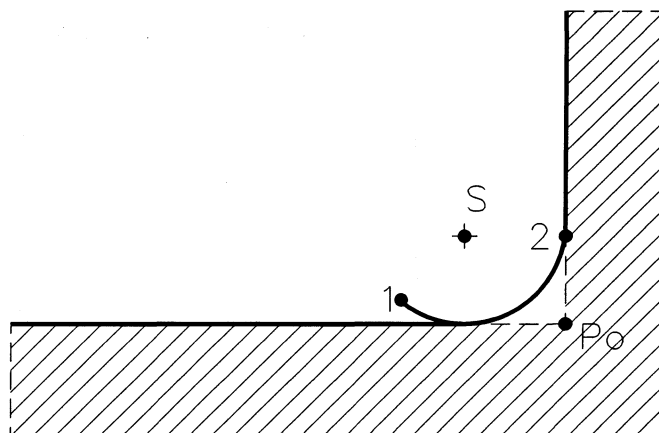
The feed-in of the contour is effected via a quarter-circle with centre S and a radius equal to the distance between S and the contour.



NB8251

$$\text{distance} = \text{factor} * \text{radius} : 100$$

A problem will arise when the start point of the feed-in circular movement is on the path of the final contour. The tool will then slightly touch the contour surface. To prevent this occurring the CNC automatically repositions the start point S.



NB 8252

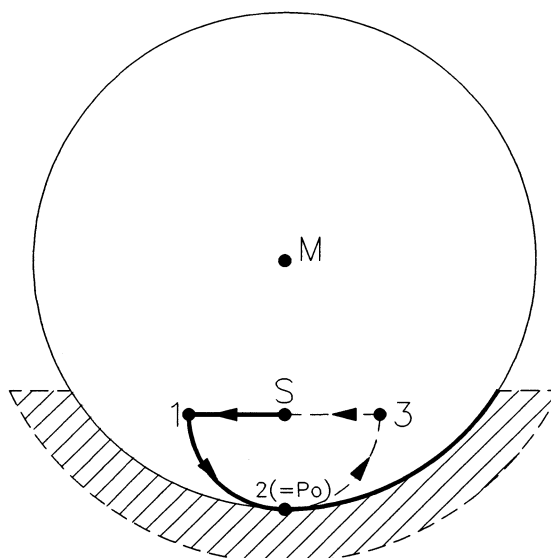
In this example the feed-in circle begins 120° before point 2.

A pocket wall can usually be machined in one step. If this is not possible, the programmer can finish the wall in several steps using datum point shifts.

After completion of a macro, the functions G0, G40 and G90 become active.

71.1.3.3 Circular pocket contour

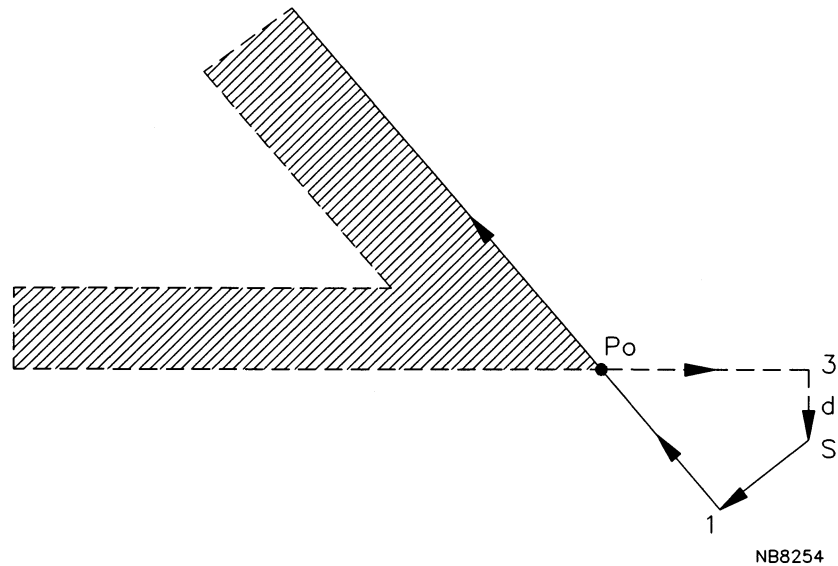
For a circular pocket contour the calculation of the feed-in point S must be changed.



NB 8253

In this case point S lies on the straight line M to Po, at a calculated distance from Po. A pocket contour will be entered at point Po (= point 2). The direction of rotation at feed-in and feed-out is the same as the programmed direction of rotation for the contour.

71.1.4 Linear feed-in movements



$$\text{distance} = \text{factor} * \text{radius} : 100$$

Note: After the macro is executed, the functions G0, G40 and G90 will be active.

71.1.5 Sequence of the macros on the machine

1. Starting points macro

This macro will run in the normal way without any restrictions.

2. Machining macro

This macro will run in the normal way as well, but some material may be left.

There are two possibilities:

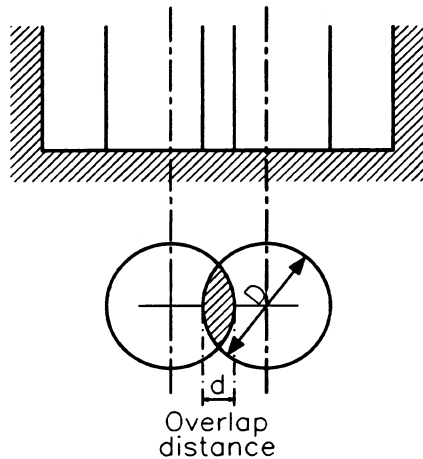
- (i). During contour-parallel milling small areas may remain unmilled when the tool changes direction between two parallel paths; this situation can also occur at contour roundings:

The solutions to these two problems are:

- a. For parallel movements - a bigger overlap between parallel movements;
- b. For contour narrowings - a tool with a smaller diameter is used.

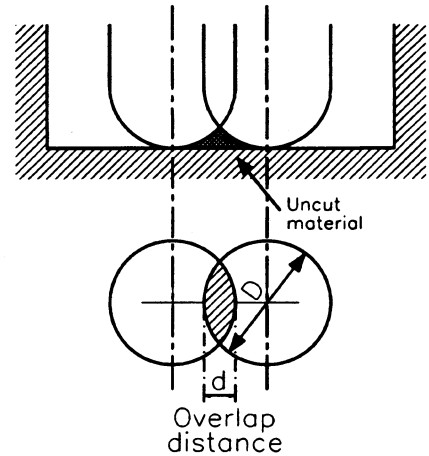
- (ii) Due to the shape of the cutting tool some material may be left on the bottom. Normally, the overlap-parameter (I-word from the G201-block) should ensure that no material is left between the paths. The greater the value of the I-address, the greater the risk that material will be left on the bottom.

A value of 50% will ensure that when a square-edge milling tool is used no material will be left in the pocket's bottom surface. However, milling tools with curved edges require different I-values because of the different radii of the curved edges.



Cylinder cutter overlap

NB 8817



Ball Cutter Overlap

NB 8818

3. Macros for finishing

The radius of the active tool will be used for radius compensation. The user should check that the cutting tool can move without damaging the pocket side. The macros will operate in the normal way.

Notes

The function G201 and the terminating function G202 must both be written in the same program or in the same macro.

Between G201 and G202 only the following functions are permitted:

G203, G204, G205, G206 with contour descriptions
G207 and G208

During description between G201 and G202 the following functions may not be used:

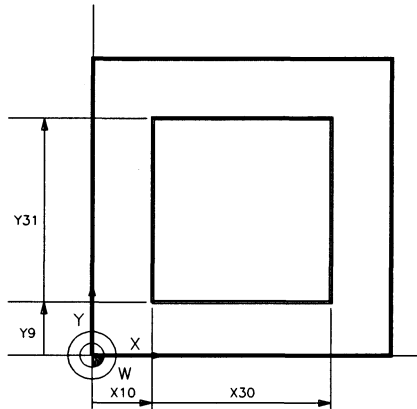
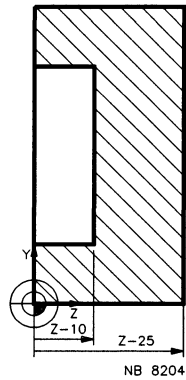
- Datum points shifts, notably (G92/G93) and stored zero offsets (G51 to G59) G54I[nr.]
- Axis rotation (G92, G93)
- Mirror image or scaling (G72, G73)
- Definition and activation of fixed cycles
- Measuring cycles
- Plane selection (G17, G18, G19)
- Point definition (G78)
- Helix interpolation
- Tool change (M06, M66, M67)
- Macro (G22) or program call (G23)
- Block and pattern repeat (G14)
- Conditional jump (G29)
- Chamfer or rounding (G11)

The use of E-parameters is not permitted for contour descriptions, or in the G201-block.

Examples of pocket descriptions

Note: For the following examples it is assumed that G17 is active.

EXAMPLE 1. Rectangle

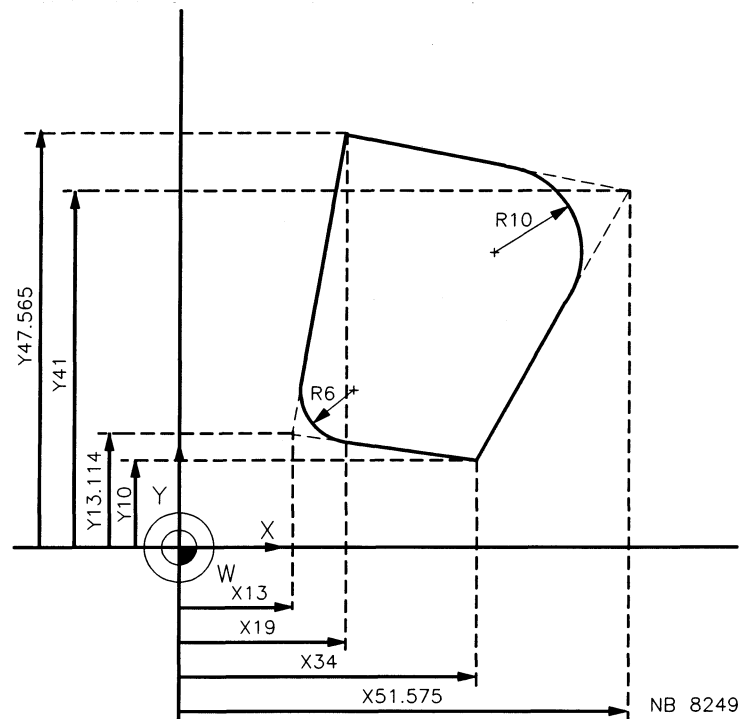


Programming can be done in two ways:

G201 N1=.. Z-10
G203 X10 Y9 Z0 N1=9800

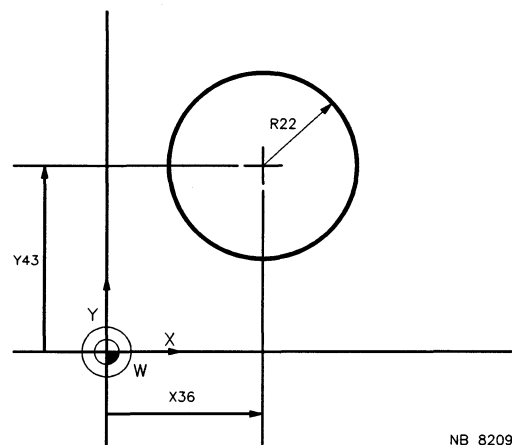
absolute	incremental
N.. X40	N.. G91
N.. Y40	N.. X30
N.. X10	N.. Y31
N.. Y9	N.. X-30
N.. G204	N.. Y-31
N.. G202	

EXAMPLE 2. Quadrangle with roundings



N.. G201 Z-10
 N.. G203 X34 Y10 Z0
 N.. G64
 N.. G1 X1=51.575 Y1=41
 N.. G3 R10
 N.. G1 X19 Y47.565
 N.. G1 X13 Y13.114
 N.. G3 R6
 N.. G1 X34 Y10
 N.. G63
 N.. G204
 N.. G202

EXAMPLE 3. Full circle



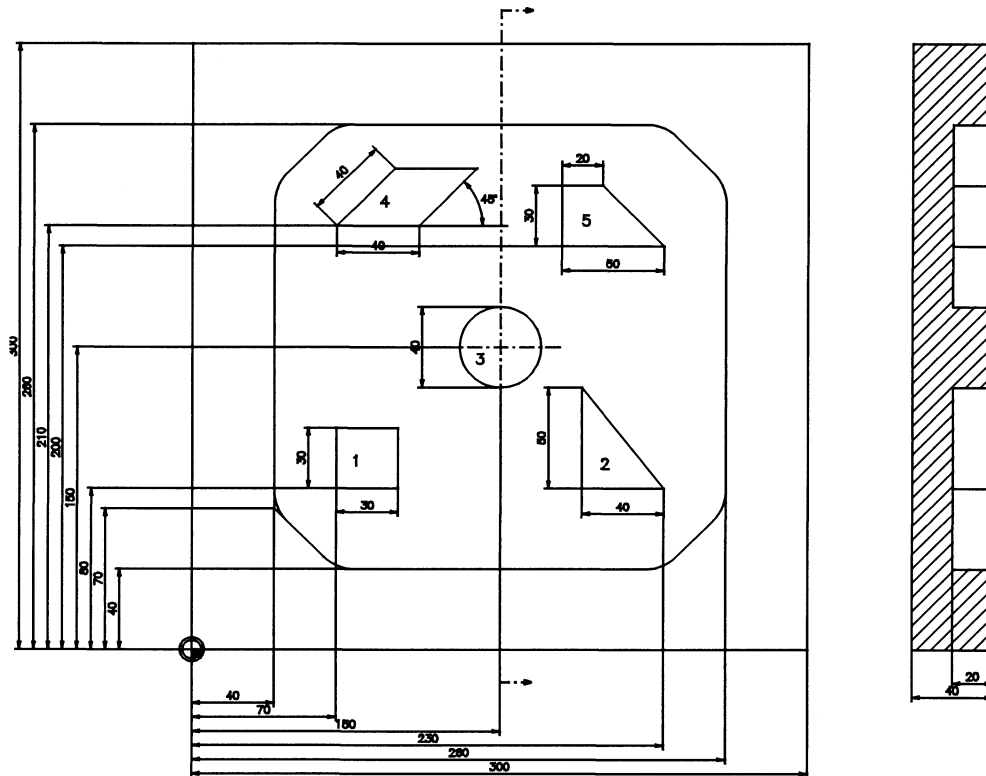
N.. G201 N1=.. Z-10
 N.. G203 X36 Y21 Z0

N.. G3 I36 J43

N.. G204

N.. G202

EXAMPLE 4. Pocket with islands



N9990

N1 G54

N2 G17

N3 G195 X-10 Y-10 Z10 I320 J320 K-60

N4 G99 X0 Y0 Z0 I300 J300 K-40

N5 G200

N6 T2 M6 (predrilling start point, drill R8)

N7 G81 Y2 Z-20 F200 S3000 M3

N8 G22 N=9992

N9 T3 M6 (clearing out the pocket, mill R10)

N10 S2500 M3

N11 G201 Y0.1 Z-20 B2 I50 K8 F200 F2=1000 N1=9991 N2=9992

:

N37 G202

Start contour pocket cycle G201

72. End contour pocket cycle G202

Purpose

End contour pocket cycle

Format

N... G202

Parameters

Non

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

Notes and Usage

The function G201 and the terminating function G202 must both be written in the same program or in the same macro.

Between G201 and G202 only the following functions are permitted:

G203, G204, G205, G206 with contour descriptions
G207 and G208

During description between G201 and G202, the following functions may not be used:

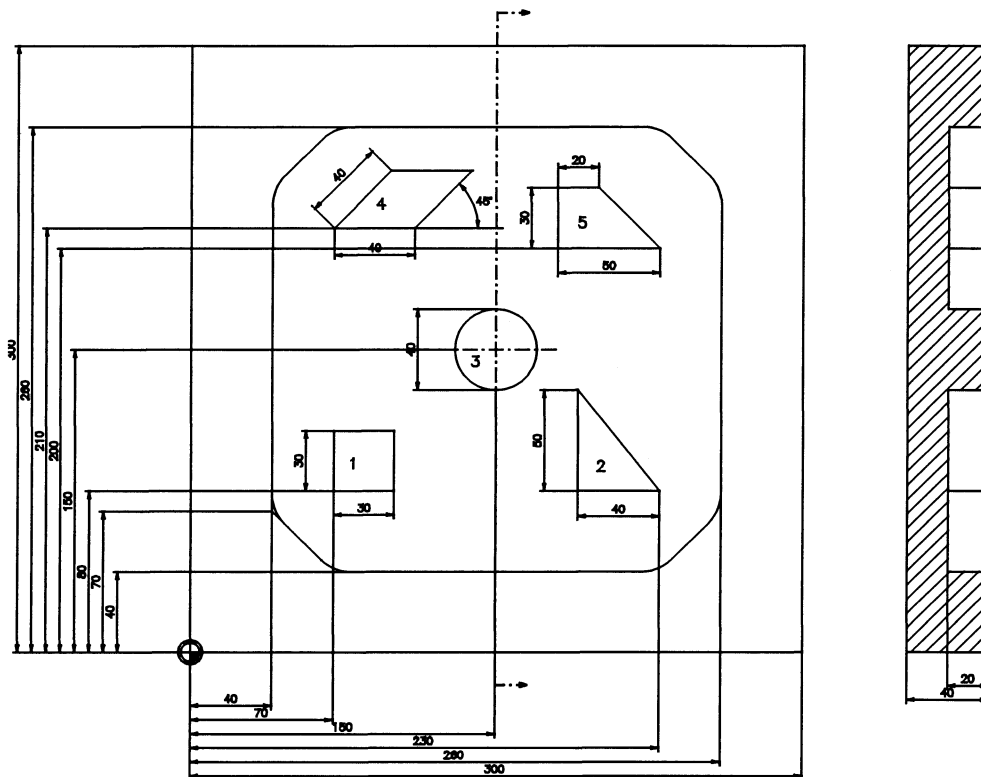
- Datum points shifts, notably (G92/G93) and stored zero offsets (G51 to G59) (G54I[nr.])
- Axis rotation (G92, G93)
- Mirror image or scaling (G72, G73)
- Definition and activation of fixed cycles
- Measuring cycles
- Plane selection (G17,G18,G19)
- Point definition (G78)
- Helix interpolation
- Tool change (M06, M66, M67)
- Macro (G22) or program call (G23)
- Block and pattern repeat (G14)
- Conditional jump (G29)
- Chamfer or rounding (G11)

Completion of the entire pocket description. After the pocket has been cut out, the partprogram will be resumed with the block following G202. Only the N-word is permitted in an G202-block.

By a G202-function the calculation for the universal pocket cycle will be stopped. After a new G200-function the calculation will be started again.

The functions G0, G40, G63 and G90 will be active after the G202-function is activated.

After the pocket description the program should be continued with an absolute position.

Example**EXAMPLE 1. Pocket with islands**

N9990

N1 G54

N2 G17

N3 G195 X-10 Y-10 Z10 I320 J320 K-60

N4 G99 X0 Y0 Z0 I300 J300 K-40

N5 G200

N6 T2 M6 (predrilling start point, drill R8)

N7 G81 Y2 Z-20 F200 S3000 M3

N8 G22 N=9992

N9 T3 M6 (clearing out the pocket, mill R10)

N10 S2500 M3

N11 G201 Y0.1 Z-20 B2 I50 K8 F200 F2=1000 N1=9991 N2=9992

N12 G203 X40 Y40 Z0 N1=9993

N13 G208 X220 Y220 I30 (pocket contour)

N14 G204

N15 G205 X100 Y80 N1=9994

N16 G208 X-30 Y30 J-1 (Island 1)

N17 G206

N18 G205 X190 Y80 N1=9995

N19 G91

N20 Y50 (Island 2)

N21 X40 Y-50

N22 G90

N23 G206

N24 G205 X150 Y130 N1=9996

N25 G2 I150 J150 (Island 3)

N26 G206

N27 G205 X110 Y210 N1=9997

N28 G208 X-40 Y40 J-1 B1=135 (Island 4)

N29 G206

N30 G205 X180 Y200 N1=9998

N31 G91

N32 Y30

N33 X20 (Island 5)

N34 X30 Y-30

N35 G90

N36 G206

N37 G202

N38 T4 M6 (clearing out the pocket, mill R8)

N39 F200 S2200 M3

N40 G22 N=9993

N41 G22 N=9994

N42 G22 N=9995

N43 G22 N=9996

N44 G22 N=9997

N45 G22 N=9998

N46 M30

(finishing)

73. Start pocket contour description G203

Purpose

Start pocket contour description

Format

N... G203 X.. Y.. Z.. {N1=..} {P..} {B1=..} {B2=..} {L2=..} {P1=..}

Parameters

X Startpoint in X
 Y Startpoint in Y
 Z Startpoint in Z
 N1= Finishing macro number
 P Point definition number
 B1= Rotation angle of contour pocket
 B2= Start point polar angle
 L2= Start point polar length
 P1= Point definition number

N1=: Number of the macro for finishing the pocket contour. A finishing macro will not be generated if the word N1=.. is not programmed.

B1=: Rotation of the pocket contour around the point from the G203-block. Islands will not be rotated.

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

Type of function

Non

Notes and Usage

The description of the pocket contour is started by stating the position of the first point: the finishing cut of the contour also starts at this point.

The starting point can be defined by absolute cartesian coordinates measured from the program zero point W. The starting point can also be programmed by using absolute polar coordinates (B2=... and L2=...). The position can also be programmed by a predefined (G78) point.

The position of the workpiece's upper surface is defined by coordinate values of the following tool axes: (G17) Z-axis, (G18) X-axis, (G19) Y-axis.

A tool axis coordinate must always be present in a G203-function's block.

G203 causes G1, G63 and G90 to become active.

The contour description can start at any point, e.g. the middle of a side's length.

The sides of the contour can be described using all the possibilities of the control to indicate linear

and circular sides.

The pocket contour must be closed; if a gap is present the CNC will automatically close that gap with a linear contour side.

Absolute or incremental coordinates are allowed in the contour description. Programming is effected via G90 or G91.

The geometric function (G64) may be used. The same conditions will apply as for programming a geometric contour. The most important is that between G64 and G63 absolute coordinates must be used exclusively.

Only axes in the main plane may be programmed. The program can start its finishing cut from the middle of a side's length; this is done by positioning the start point of the pocket cycle description at the middle of the side.

The program starts its finishing cut at the start point of the contour description. The finishing cycle will follow the same sequence as the contour description.

Only the following functions are permitted between G203 and G204, or between G205 and G206:

G1, G2, G3, G208
G63, G64
G90, G91

The G1, G2/G3 movements are limited to the main plane. Tool-axis and rotary axis coordinates are not permitted.

The G63/G64 and G90/G91 functions must all be programmed in separate blocks eg:

```

N10    G64
N20          X...    Y... :   This is acceptable.

N10    G64    X...    Y... :   This is not acceptable.

```

The associated pair of functions G203/G204 and G205/G206 must be written in the same program or the same macro.

The first point of a contour description must be given in a G203-block.

Defined points may be used in a contour, but a point definition (G78) is not allowed. The points must have been defined prior to the G200-block of the pocket.

The bottom of the pocket must be parallel to the machining plane, i.e. XY-plane (G17), XZ-plane (G18) or YZ-plane (G19). A slanted or a curved bottom plane is not allowed.

The sides of the pocket must be perpendicular to the bottom plane.

Two parts of the same pocket may not intersect, or be tangent, to each other. However, two different pockets may intersect each other.

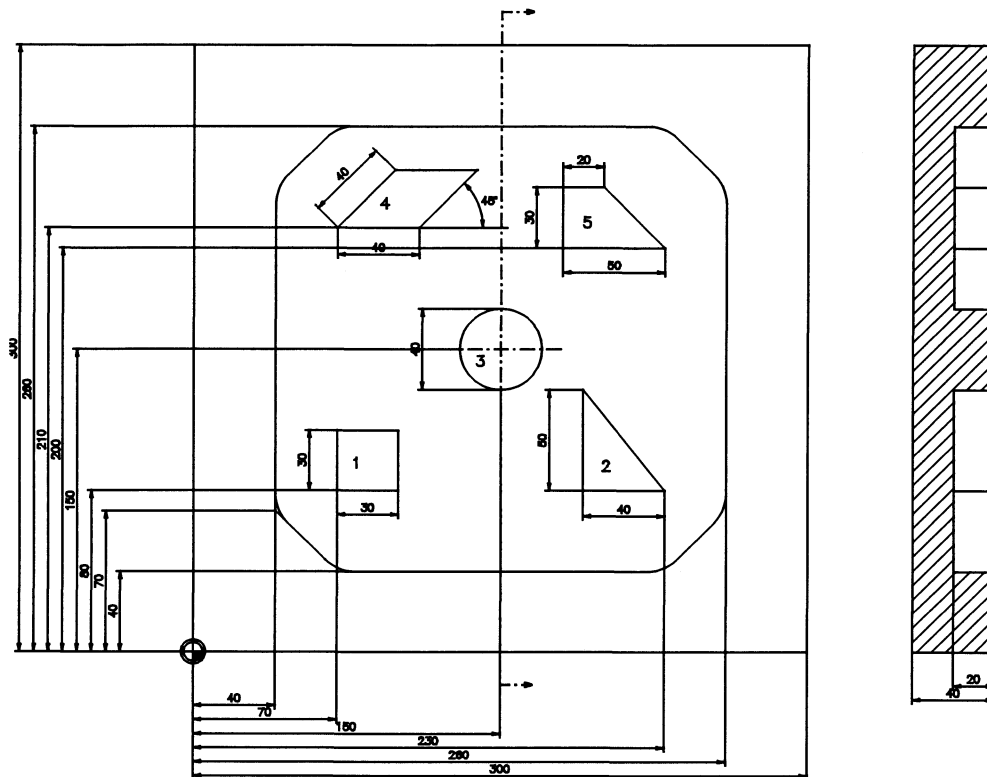
During finishing the programmer should make sure that the tool diameter is selected smaller than the distance to the narrowest part in pocket of the workpiece. The control system is unable to identify contours damaged during finishing.

The rules which apply to G1 and G2/G3 movements also apply to the same types of movement used within pocket cycles.

When the (G64) Geometry function is active, only absolute (G90) coordinates can be used and only one predefined (G78) point can be present in a program block.

Example

EXAMPLE 1. Pocket with islands



N9990

N1 G54

N2 G17

N3 G195 X-10 Y-10 Z10 I320 J320 K-60

N4 G99 X0 Y0 Z0 I300 J300 K-40

N5 G200

N6 T2 M6 (predrilling start point, drill R8)

N7 G81 Y2 Z-20 F200 S3000 M3

N8 G22 N=9992

N9 T3 M6 (clearing out the pocket, mill R10)

N10 S2500 M3

N11 G201 Y0.1 Z-20 B2 I50 K8 F200 F2=1000 N1=9991 N2=9992

N12 G203 X40 Y40 Z0 N1=9993

N14 G204

N37 G202

Start pocket contour description G203

74. End pocket contour description G204

Purpose

End pocket contour description

Format

N... G204

Parameters

Non

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

Type of function

Non

Notes and Usage

G204 is only allowed to be programmed between G201 and G202.

Only the following functions are permitted between G203 and G204:

G1, G2, G3, G208

G63, G64

G90, G91

The G1, G2/G3 movements are limited to the main plane. Tool-axis and rotary axis coordinates are not permitted.

The G63/G64 and G90/G91 functions must all be programmed in separate blocks eg:

N10 G64

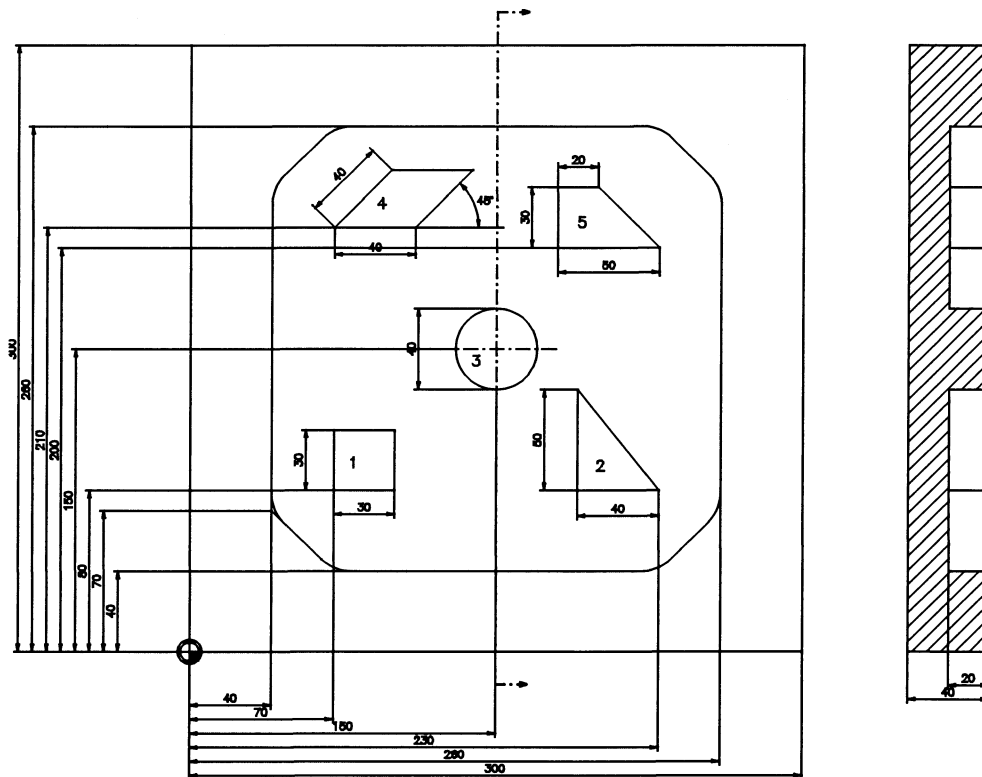
N20 X... Y... : This is acceptable.

N10 G64 X... Y... : This is not acceptable.

The associated pair of functions G203/G204 must be written in the same program or the same macro.

Example

EXAMPLE 1. Pocket with islands



N9990

N1 G54

N2 G17

N3 G195 X-10 Y-10 Z10 I320 J320 K-60

N4 G99 X0 Y0 Z0 I300 J300 K-40

N5 G200

N6 T2 M6 (predrilling start point, drill R8)

N7 G81 Y2 Z-20 F200 S3000 M3

N8 G22 N=9992

N9 T3 M6 (clearing out the pocket, mill R10)

N10 S2500 M3

N11 G201 Y0.1 Z-20 B2 I50 K8 F200 F2=1000 N1=9991 N2=9992

N12 G203 X40 Y40 Z0 N1=9993

N13 G208 X220 Y220 I30 (pocket contour)

N14 G204

N37 G202

75. Start island contour description G205

Purpose

Start island contour description

Format

N... G205 X.. Y.. Z.. {N1=..} {P..} {B1=..} {B2=..} {L2=..} {P1=..}

Parameters

X Startpoint in X
 Y Startpoint in Y
 Z Startpoint in Z
 N1= Finishing macro number
 P Point definition number
 B1= Rotation angle of island contour
 B2= Start point polar angle
 L2= Start point polar length
 P1= Point definition number

N1=: Number of the macro for finishing the pocket contour. A finishing macro will not be generated if the word N1=.. is not programmed.

B1=: Rotation of the pocket contour around the point from the G203-block. Islands will not be rotated.

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

Type of function

Non

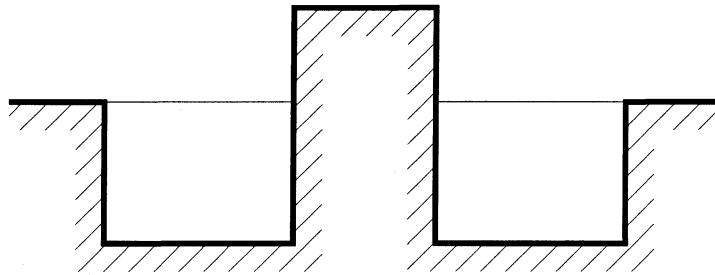
Notes and Usage

The contour of an island is described in the same way as the contour of a pocket. The description begins with G205 and the absolute starting position of the island.

The absolute position is described with either:

- cartesian coordinates
- polar coordinates
- a defined point.

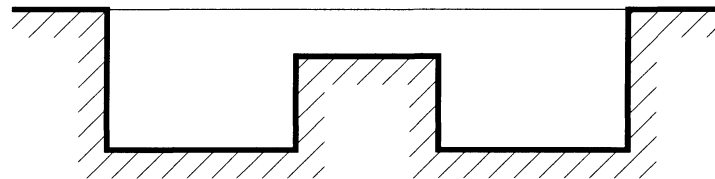
Programming of the tool axis is not allowed. The CNC assumes that the top surface of the island coincides with the top surface of the pocket.



NB 8211

If the island rises above the upper surface of the pocket, the B-word from the G201-block must be used to prevent a collision between cutting tool and workpiece during a movement from one starting point to the other.

If the upper surface of the island lies below the upper surface of the pocket, the part programmer must see to it that, after the pocket has been cut, the island is brought to the correct level.

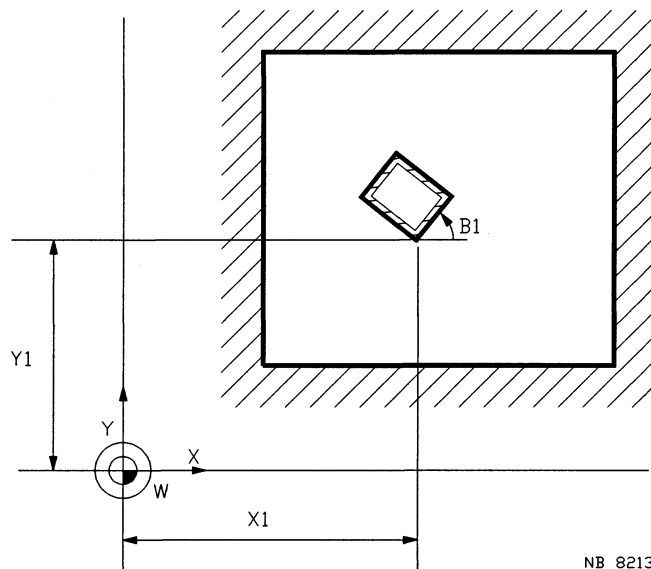


NB 8212

G205 causes G1, G63 and G90 to become active.
The contour description is terminated with G206.

For more information about island contour description see G203.

ROTATION OF AN ISLAND AROUND THE STARTING POINT



NB 8213

The word B1=.. from the G205-block indicates that the island is rotated around its starting point (X1;Y1) through the programmed angle.

The programming will be:

```
G205 X (=X1)          Y (=Y1)          B1= (=B1)      N1=..
:      \
:      >      island contour
:      /
G206
```

Notes

Only the following functions are permitted between G205 and G206:

G1, G2, G3, G208

G63, G64

G90, G91

The G1, G2/G3 movements are limited to the main plane. Tool-axis and rotary axis coordinates are not permitted.

The G63/G64 and G90/G91 functions must all be programmed in separate blocks eg:

N10 G64

N20 X... Y... : This is acceptable.

N10 G64 X... Y... : This is not acceptable.

The associated pair of functions G205/G206 must be written in the same program or the same macro.

The contour of an island must be closed. See GENERAL REMARKS

Two island may not intersect with each other or be tangent.

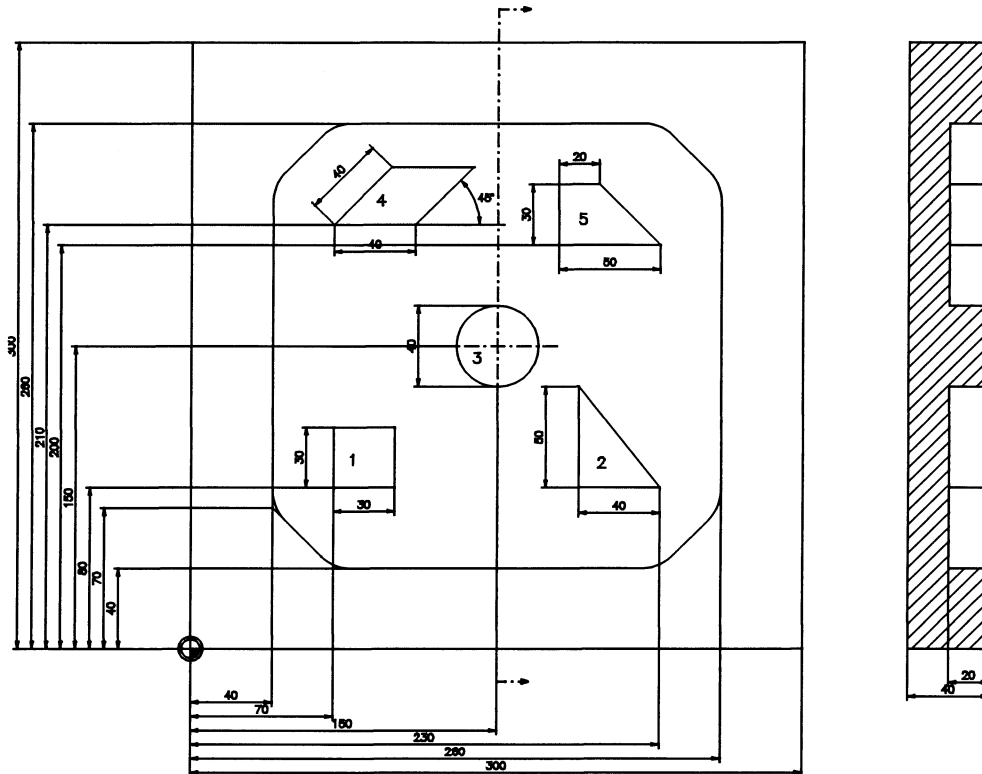
Islands must be situated in the pocket and may not intersect with, or be tangent to, the pocket sides.

The sides of an island must be perpendicular to the bottom plane.

An island may not be enclosed by another island.

Example

EXAMPLE 1. Pocket with islands



N9990

N1 G54

N2 G17

N3 G195 X-10 Y-10 Z10 I320 J320 K-60

N4 G99 X0 Y0 Z0 I300 J300 K-40

N5 G200

N6 T2 M6 (predrilling start point, drill R8)

N7 G81 Y2 Z-20 F200 S3000 M3

N8 G22 N=9992

N9 T3 M6 (clearing out the pocket, mill R10)

N10 S2500 M3

N11 G201 Y0.1 Z-20 B2 I50 K8 F200 F2=1000 N1=9991 N2=9992

N12 G203 X40 Y40 Z0 N1=9993

N13 G208 X220 Y220 I30 (pocket contour)

N14 G204

N15 G205 X100 Y80 N1=9994

N16 G208 X-30 Y30 J-1 (Island 1)

N17 G206

N18 G205 X190 Y80 N1=9995

N19 G91

N20 Y50 (Island 2)

N21 X40 Y-50

N22 G90

N23 G206

N24 G205 X150 Y130 N1=9996

N25 G2 I150 J150 (Island 3)

N26 G206

N27 G205 X110 Y210 N1=9997

N28 G208 X-40 Y40 J-1 B1=135 (Island 4)

N29 G206

N30 G205 X180 Y200 N1=9998

N31 G91

N32 Y30

N33 X20 (Island 5)

N34 X30 Y-30

N35 G90

N36 G206

N37 G202

N38 T4 M6 (clearing out the pocket, mill R8)

N39 F200 S2200 M3

N40 G22 N=9993

N41 G22 N=9994

N42 G22 N=9995

N43 G22 N=9996

N44 G22 N=9997

N45 G22 N=9998

N46 M30

(finishing)

76. End pocket contour description G206

Purpose

End pocket contour description

Format

N... G206

Parameters

Non

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

Type of function

Non

Notes and Usage

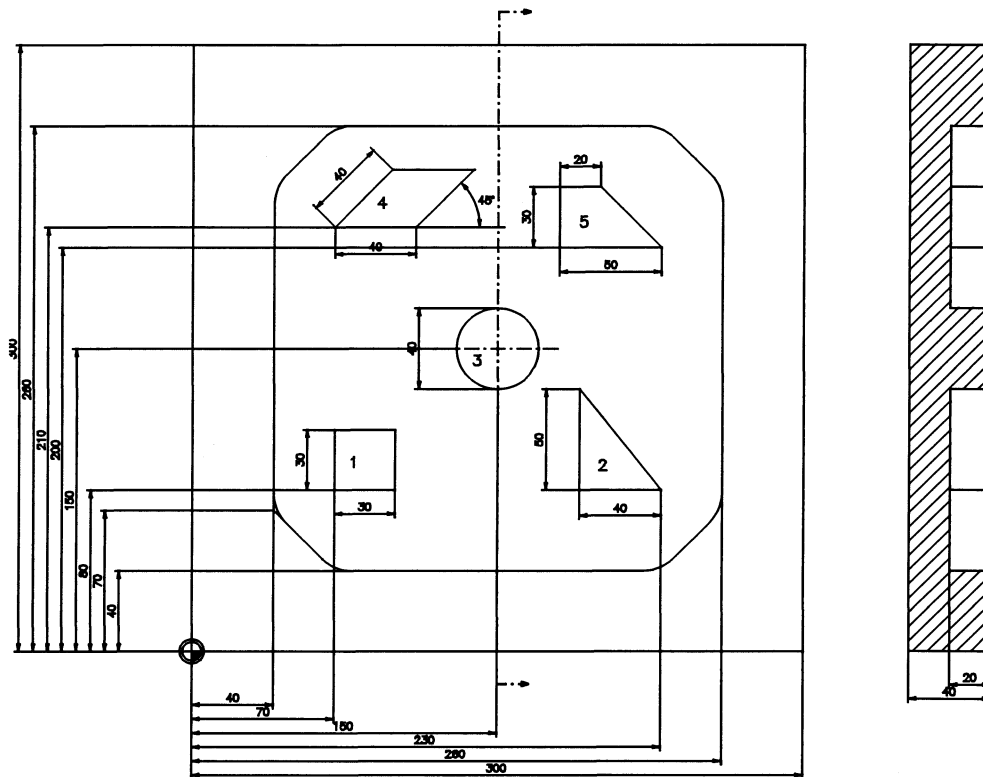
Only the following functions are permitted between G205 and G206:

G1, G2, G3, G208

G63, G64

G90, G91

The associated pair of functions G205/G206 must be written in the same program or the same macro.

Example**EXAMPLE 1. Pocket with islands**

N9990

N1 G54

N2 G17

N3 G195 X-10 Y-10 Z10 I320 J320 K-60

N4 G99 X0 Y0 Z0 I300 J300 K-40

N5 G200

N6 T2 M6 (predrilling start point, drill R8)

N7 G81 Y2 Z-20 F200 S3000 M3

N8 G22 N=9992

N9 T3 M6 (clearing out the pocket, mill R10)

N10 S2500 M3

N11 G201 Y0.1 Z-20 B2 I50 K8 F200 F2=1000 N1=9991 N2=9992

N12 G203 X40 Y40 Z0 N1=9993

N13 G208 X220 Y220 I30 (pocket contour)

N14 G204

N15 G205 X100 Y80 N1=9994

N16 G208 X-30 Y30 J-1 (Island 1)

N17 G206

N18 G205 X190 Y80 N1=9995

N19 G91

N20 Y50 (Island 2)

N21 X40 Y-50

N22 G90

N23 G206

N24 G205 X150 Y130 N1=9996

N25 G2 I150 J150 (Island 3)

N26 G206

N27 G205 X110 Y210 N1=9997

N28 G208 X-40 Y40 J-1 B1=135 (Island 4)

N29 G206

N30 G205 X180 Y200 N1=9998

N31 G91

N32 Y30

N33 X20 (Island 5)

N34 X30 Y-30

N35 G90

N36 G206

N37 G202

N38 T4 M6 (clearing out the pocket, mill R8)

N39 F200 S2200 M3

N40 G22 N=9993

N41 G22 N=9994

N42 G22 N=9995

N43 G22 N=9996

N44 G22 N=9997

N45 G22 N=9998

N46 M30

(finishing)

77. Call island contour macro G207

Purpose

Programs the same island contour in another place

Three possibilities are available:

1. The same island occurs in another place within the same pocket contour.
2. The same island contour occurs in another pocket contour.
3. The same island contour occurs in another program.

By including the island contour in a macro the three possibilities can be treated in the same way.

Format

N... G207 X.. Y.. Z.. N=.. N1=..

Parameters

X Shift along in X
Y Shift along in Y
Z Shift along in Z
N= Macro with islandcontour
N1= Finishing macro number

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

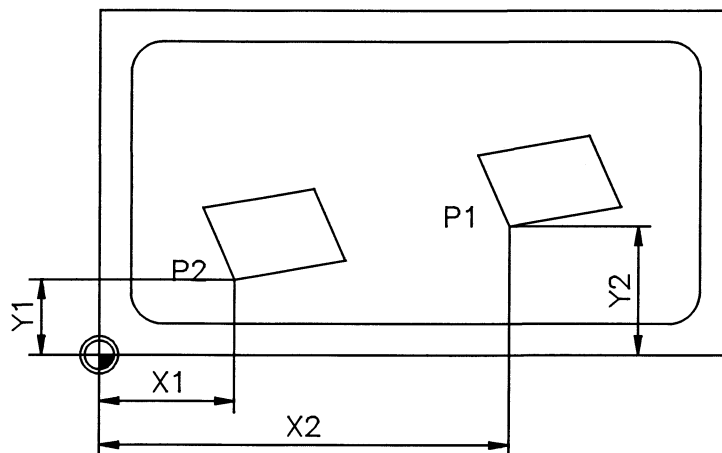
Type of function

Non

Notes and Usage

Geometric programmed contours (G64...G63) within a G207 macro (pocketcycli) may function incorrect because the macro-offset is only taken into account at the start point. Programmers using pocket cycli are advised to check the pocket macro's on this phenomenon and if so to incorporate the macro concerned into the part program in order to overcome the problem.

Example



The macro of the island contour will then be:

```
%MM
N9xxx G205 X.. Y.. N1=..

N1                                \
:                                > island contour
:
:
:                                /
N..
N G206
```

Here, N9xxx is the identification of the macro.

The macro is called with the function G207.

N.. G201

N.. G207 N=9xxx N1..

N.. G207 N=9xxx X=(X1-X2) Y=(Y1-Y2) N1=..

N.. G202

Explanation:

1: Island whose contour is programmed as a macro.

P1: Starting point of the contour description (G205-block).

2: Desired position of the island.

P2: Starting point of the shifted contour.

X...: Distance parallel to X-axis from P1 to P2.

Y...: Distance parallel to Y-axis from P1 to P2.

The distance carries a sign, as with incremental programming.

78. Quadrangle contour description G208

Purpose

To describe a regular rectangle as a pocket or island contour.

The G208 can be used to easily program a regular rectangle, more in particular a rectangle or parallelogram, as a contour.

Format

N... G208 X.. Y.. Z.. {I..} {J..} {R..} {B1=..}

Parameters

X Length in X
Y Length in Y
Z Length in Z
I Chamfer length
J J1:climb / J-1:conventional
R Rounding radius
B1= Angle quadrangle contour

Associated functions

G200...G208

For general description about universal pocket cycles see G200.

Type of function

Non

Notes and Usage

N: Block number

X and Y: These words state distances along the two main-plane axes. These distances are measured from the start point stated by the G203-function. The sign of each word states the direction in which a distance was measured. A '+' distance is measured in a positive axis direction and a '-' distance measured in a negative axis direction.

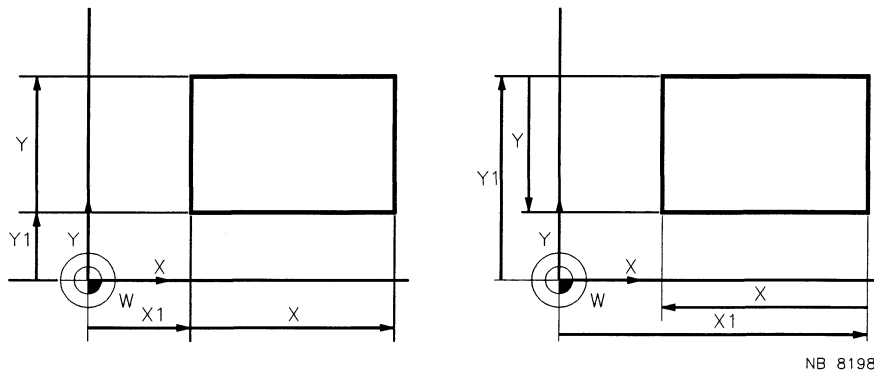
The bottom of the pocket must always be parallel to the main-plane.

Selection of plane

The default plane used in this manual is the XY-plane which is selected by the G17-function: two other main planes are available. The XZ-plane is selected by the G18-function and the YZ-plane by the G19-function.

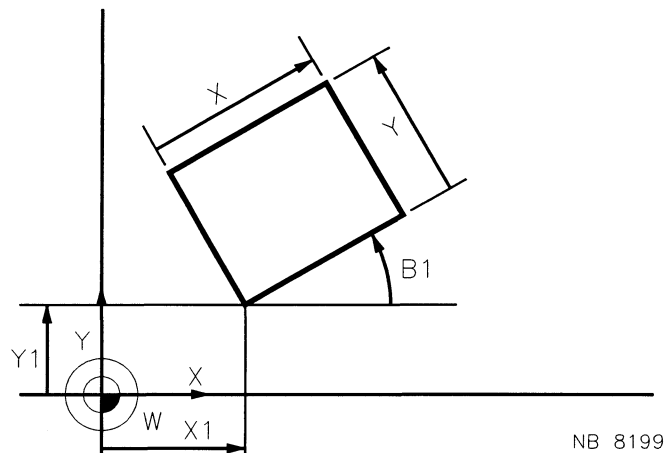
Examples

EXAMPLE 1. Rectangle



If the sides of a rectangle are parallel to the axes, the X and Y coordinates are equally parallel to the axes.

If the sides are not parallel to the X and Y-axis, G208 is programmed as if the sides are parallel. The G203- block contains the rotation angle (B1=...).



Programming will be:

G203 X (=X1) Y (=Y1) Z (=Z1) B1= (=B1)

G208 X (=X) Y (=Y)

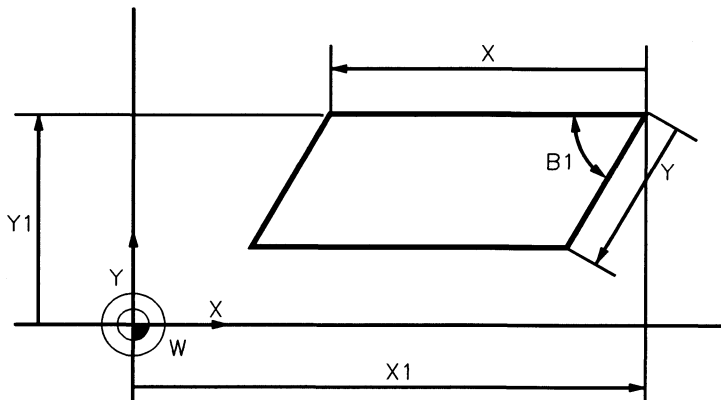
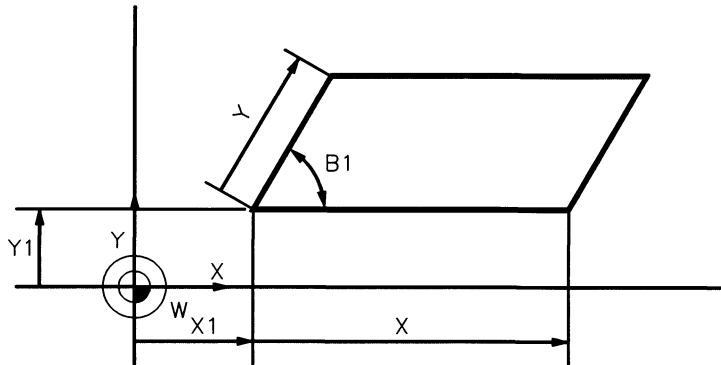
G204

EXAMPLE 2. Parallelogram

To program a parallelogram the lengths of both sides and the angle between them at the startpoint are stated.

Here:

B1=: The angle in degrees and decimal fractions of degrees
 ($0^\circ < B1 < 180^\circ$).
 The angle carries no sign.
 The default value for B1=.. is 90 degrees, ie. a rectangle.



NB 8200

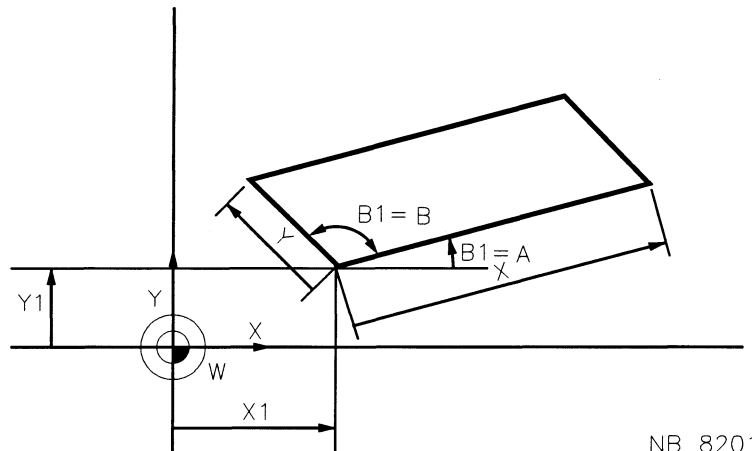
Programming will be:

G203 X (=X1) Y (=Y1) Z (=Z1)

G208 X (=X) Y (=Y) B1= (=B1)

G204

If one side is not parallel to the X-axis, programming will be the same. The G203-block contains the rotation angle.



NB 8201

The programming will be:

G203 X (=X1) Y (=Y1) Z (=Z1) B1= (=A)

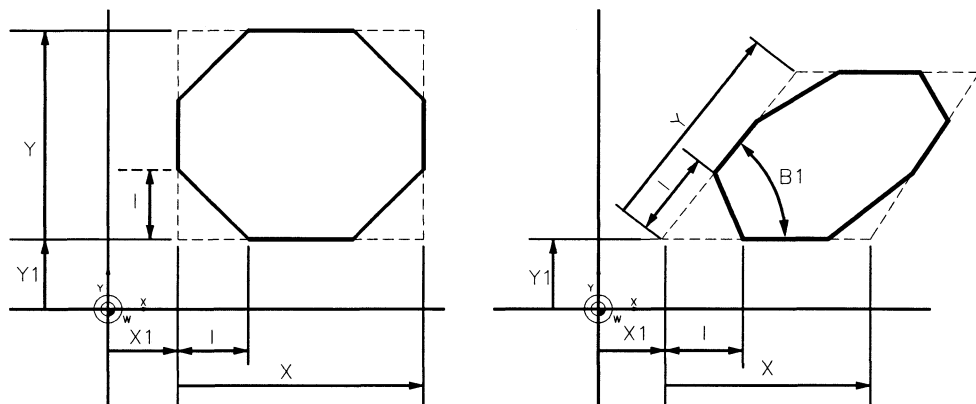
G208 X (=X) Y (=Y) B1= (=B)

G204

EXAMPLE 3. One chamfer

For both the rectangle and the parallelogram a bevel or phase can be added.

I-word: The width of the chamfer.
The I-word carries no sign.
The chamfer is symmetrically arranged around the corner.



NB 8202

The programming will be:

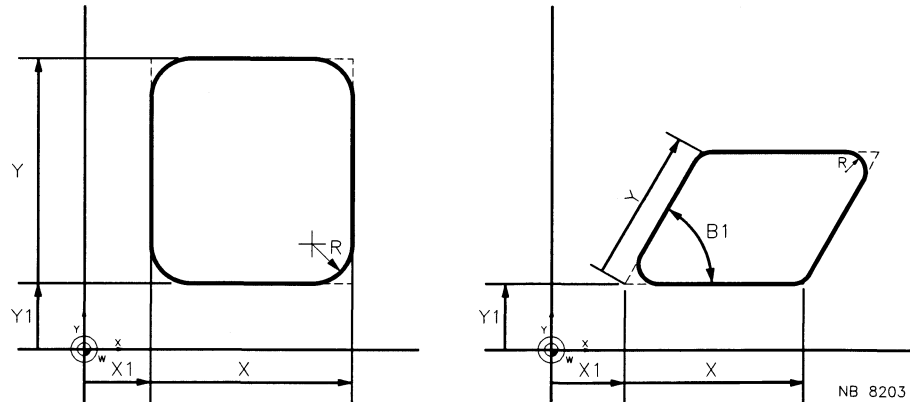
G203 X (=X1) Y (=Y1) Z (=Z1)

G208 X (=X) Y (=Y) B1= (=B1) I (=I)

G204

EXAMPLE 4. Rounding

R-word: The radius of the rounding. The radius does not have a '+/-' sign.



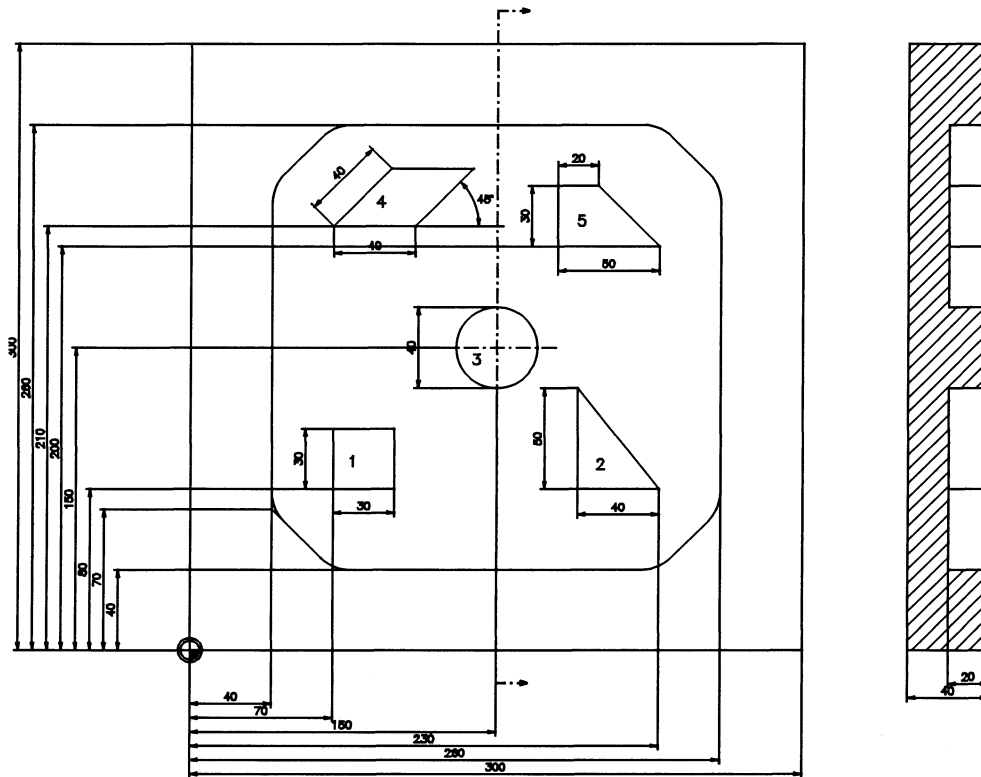
The programming will be:

```
G203 X (=X1) Y (=Y1) Z (=Z1)
G208 X (=X) Y (=Y) B1= (=B1) R (=R)
G204
```

Either the I-word or the R-word can be programmed: the two words cannot be used at the same time, if this is done, an 'invalidation contour description' message (P144) is generated.

Example

EXAMPLE 5. Pocket with islands



N9990

N1 G54

N2 G17

N3 G195 X-10 Y-10 Z10 I320 J320 K-60

N4 G99 X0 Y0 Z0 I300 J300 K-40

N5 G200

N6 T2 M6 (predrilling start point, drill R8)

N7 G81 Y2 Z-20 F200 S3000 M3

N8 G22 N=9992

N9 T3 M6 (clearing out the pocket, mill R10)

N10 S2500 M3

N11 G201 Y0.1 Z-20 B2 I50 K8 F200 F2=1000 N1=9991 N2=9992

N12 G203 X40 Y40 Z0 N1=9993

N13 G208 X220 Y220 I30 (pocket contour)

N14 G204

N15 G205 X100 Y80 N1=9994

N16 G208 X-30 Y30 J-1 (Island 1)

N17 G206

N18 G205 X190 Y80 N1=9995

N19 G91

N20 Y50 (Island 2)

N21 X40 Y-50

N22 G90

N23 G206

N24 G205 X150 Y130 N1=9996

N25 G2 I150 J150 (Island 3)

N26 G206
 N27 G205 X110 Y210 N1=9997
N28 G208 X-40 Y40 J-1 B1=135 (Island 4)
 N29 G206
 N30 G205 X180 Y200 N1=9998
 N31 G91
 N32 Y30
 N33 X20 (Island 5)
 N34 X30 Y-30
 N35 G90
 N36 G206
 N37 G202
 N38 T4 M6 (clearing out the pocket, mill R8)
 N39 F200 S2200 M3
 N40 G22 N=9993
 N41 G22 N=9994
 N42 G22 N=9995
 N43 G22 N=9996
 N44 G22 N=9997
 N45 G22 N=9998
 N46 M30

(finishing)

79. Programming error messages G300

Purpose

Setting error messages during the execution of universal programs or macros.

Format

N... G300 D...

Parameter

D Error identification
 number

Notes and usage

An error message may be given during program execution.

The error messages only cover the existing P-errors (refer to INSTALLATION MANUAL).

Examples

Example : Setting an error message if a programmed angle is not allowed.

```
N9999 (Macro for calculation of table rotations)
N7 (input parameter:)
N11 (E4: phi)
N100
N110 G29 I1 E30 N=180 E30=(E4>360)
N120 G29 I1 E30 N=210 E30=(E4<0)
N150 G29 I1 E30 N=290 E30=1
N160
N170 (error message: phi>360)
N180 G300 D190 (programmed value > maximum value)
N190
N200 (error message: phi<0)
N210 G300 D191 (programmed value < minimum value)
N220
N290
```

Explanation:

```
N11 : E4 is the input value for angle phi
N110: Compare if E4 > 360 degrees. If so, jump to N180
N120: Compare if E4 < 0 degree. If so, jump to N210
N150: Jump to 290 (0 <= E4 <= 360 degrees)
N180: Error message: programmed value > maximum value
      Program should be ended and a modified E4 be entered
N210: Error message: programmed value < minimum value
      Program should be ended and a modified E4 be entered
N290: End program
```


80. Error in a program or macro G301

Error in a program or macro block.

Parameter

non

Notes and usage

When the controller retrieves a program block or macro block and discovers an error it activates G301

Function G301 can only be active in an error stopped program or macro.

The error texts are O errors. (refer to INSTALLATION MANUAL).

Example

The program on harddisk is as follows:(program is made with a MC84=0)

```
N9999(Program)
N1 G17
N2 G57
N3 T1 M6
N4 F200 S1000 M3
..
N99 M30
```

Error stops program in RAM.

Zero point shift extension MC84 > 0 is active.

```
N9999(ERR*)(Program ...)
N1 G17
N2 G301 (O138 G57)
N3 T1 M6
N4 F200 S1000 M3
..
N99 M30
```

Explanation:

N2: G301 explains that the program is false. G57 must be G54 I3

Note: The false program can be activated. When passing the block G301 the controller stops and gives the following error text P33(Modify block converted to connect). The block containing G301 must be changed before restarting.

81. Calling machine constant values G322

Purpose

To read out a machine constant value and store it in the appropriate E-parameters.

Format

N... G322 C.. N1=...

Parameter

N1= Machine constant number
C E-parameter

Notes and usage

READING OUT A MACHINE CONSTANT WITHOUT VALUE

When invisible addresses are read from the machine constant table, the E-parameter remains unchanged.

Examples

Example : Read out machine constant 103 and store value in E-parameter 10.

N... G322 N1=103 C10

E10 contains the number of machine constant 103

Example : Universal program blocks which can be used for both zero point table types.

N50 G322 N1=84 C10
N60 G29 E1 N=90 E1=E10>0
N70 G150 N1=57 X7=E1 Z7=E6
N80 G29 E1 N=100 E1=1
N90 G150 N1=54.3 X7=E1 Z7=E6
N100 ..

Explanation:

N50 : Machine constant 84 is set in E10
N60 : Compare if MC84 > 0. If so, jump to N90
N70 : Store the zero point shift table ZO.ZO
N80 : Jump to N100
N90 : Store the zero point shift table ZE.ZE

82. Calling actual axes-positions values G326

Purpose

To read out the actual axes-positions values and store it in the appropriate E-parameters.

Format

N... G326 {X7=..} {Y7=..} {Z7=..} {A7=..} {B7=..} {C7=..}

Parameter

X7= E-par. for measured value in X
Y7= E-par. for measured value in Y
Z7= E-par. for measured value in Z
A7= E-par. for measured value in A
B7= E-par. for measured value in B
C7= E-par. for measured value in C

Notes and usage

This function may only be use in programs or macros.

READING OUT OF NOT EXISTENT AXES

When an axis not exist the contents of the E-parameter remains unchanged.

READING OUT BY GRAPHICAL SIMULATION

By graphical simulation only the X,Y and Z can be read out. The E-parameters for the rotating axes stays zero.

Examples

Example 1:

Read out actual axes-position von X,Y and Z and store the values in E-parameters 20, 21 and 22.

N... G326 X7=20 Y7=21 Z7=22

E20 contains the actual X-axis-position.

Example 2:

Program continuation after an universal pocketcycli.

```
N30 G202
N40 G326 X7=20 Y7=21
N50 G29 E1 N=90 E1=E20>100
N60 G29 E1 N=90 E1=E20<-100
N70 G0 X-110
N80 G0 Y 100
N90 ..
```

Explanation:

N30 : End pocketcycli
N40 : Unknown actual End-position von X and Y
N50 : Actual X-position >100, then jump to N90
N60 : Actual X-position <-100, then jump to N90
N70 : G0 movement to X-110, if the actual X-position is be situated between 100 and -100.
On this manner for example an obstacle can be rounded.
N80 : Further turn aside movement
N90 :

83. Look Ahead Feed (LAF) function (starting from V320)

83.1 Introduction

The Look Ahead Feed function is used to carry out a precalculation on the programmed tool path, while taking account of the dynamics of all axes involved. The tool path speed is then adjusted to achieve the highest contour accuracy at the highest possible speed. However, the programmed feed is never exceeded.

Taking account of the programmed feed and the actual feed override setting, special high-performance algorithms ensure a homogenous feed for fast finishing times.

The execution speed of CAD-generated programs is substantially increased.

The user need not look at anything else when working with Look Ahead Feed. This function cannot be influenced.

Only the G28 function was changed. The addresses for feed limitation were cancelled (see G27/628, starting from V320). Existing programs need not be adapted, they can be run as usual. These functions are ignored during machining operations. The machining operation may, however, continue.

During Look Ahead Feed the end point and centre point of a circle should match within 64 μm . In this case, the centre point is automatically corrected. As opposed to V310, there is no "compensation movement" at the end point. An error message is given if the end point and centre point do not match within 64 μm . This also applies to helix interpolation.

83.2 Detailed specification

1) Types of interpolation

The LAF function is active during:

- G0 Rapid traverse, including infeed movements
- G1 Feed movement
- G2, G3 Circle, including helix interpolation

The LAF function is inactive during:

- G6 Splines
- G74 Positioning movements
- G84 Threading
- G145, G45, G46 Measuring movement
- G182 Cylinder jacket interpolation with all permitted movements
- PLC-controlled axis movements (Home position)
- Auxiliary axis movements

Circular interpolation

- The circular accuracy achieved with LAF at higher speeds is higher than that with V3.10. This is true of circles made with G2/G3 and with cycles.

2) Previous contour accuracy functions (G28 function)

The following G28 programming functions are no longer active:

- I3=2 G1,G2,G3 with corner release distance (MC136)
- I3=3 G1 with programmable contour accuracy, (MC137) or I7=(0-10000)[μ m]
- I4=2 G0 with corner release distance (MC136)
- I6=1 G2,G3 with feed limitation (MC135)

These functions are ignored during machining. The machining operation may, however, continue.

4) New error messages

P300 LAF: End point not on circle

Circle end point deviation exceeds 64 μ m.

The following applies additionally to a cylinder: 100 mm < R-cylinder < 10 m

Remedy: The end point should be defined more accurately.

P301 LAF: SW limit switch approached

The programmed path will go beyond the limit switches or outside + or - 100 m. In the case of straight lines this error is generated at the beginning of the wrongly programmed block. In the case of circles, it depends on the circular form and speed.

Remedy: The path should be defined within the possible range.

P302 No interpolation axis

The wrong axes have been defined for the selected type of interpolation:

- No two main axes for circular interpolation
- No rotary axis for cylinder jacket interpolation

Remedy: The missing axis should be defined.

Example

```
N14803 (TEST PROGRAM)
N1 (MAX. RUNTIME 24 MINUTES)
N1 (LAST BLOCK NUMBER N120666)
N6 G90
N12 G40
N14 M9
N16 T2 M67
N18 (TOLERANCE 0.01)
N20 (SPHERICAL CUTTER D 8 mm, OFFSET 0.0 )
N24 F15000 S16000 M3
N26 G0 X230 Y-14.8 Z10
N28 G0 Z42.5
N30 G1 Z32.5
N32 X210.16

N120664 G0 Z100
N120666 M30
```

83.3 General theoretical description

LAF = Look Ahead Feed. Function to prepare the tool path for checking acceleration along the tool path. With this acceleration check the maximum acceleration torque of the drives involved is not exceeded.

CAD program

NC program generated by a Computer Aided Design (CAD) System. An NC program mostly consists of many short linear blocks. The LAF function has distinct advantages, because the new LAF acceleration check is capable of looking several blocks ahead.

Inpod

In Position Delayed. Condition of the control system whereby the exact position is approached at the end of an NC block. The machine stops at this position for a particular period of time (which can be set).

The LAF functions have an effect on accuracy. Accuracy cannot be programmed in the part program.

Corner error

The accuracy of the LAF is mainly influenced by the bell-shaped filter time. A corner error of approx. 10 μm is achieved with a bell-shaped filter time of 15 ms (MC3x90 = 15). This corner error is directly proportional to the bell-shaped filter time, but independent of the corner size or the maximum axis acceleration.

Bell-shaped filter

The axes are controlled by different dynamics inside and outside LAF. When LAF is active, the bell-shaped filter time may be reduced. A higher accuracy can then also be achieved during LAF operation.

Circular accuracy

For circular radius errors the effect of the bell-shaped filter time is greater than that of the LAF machine constants. A circular radius error of 9 μm maximum is achieved with a bell-shaped filter time of 15 ms (MC3x90 = 15).

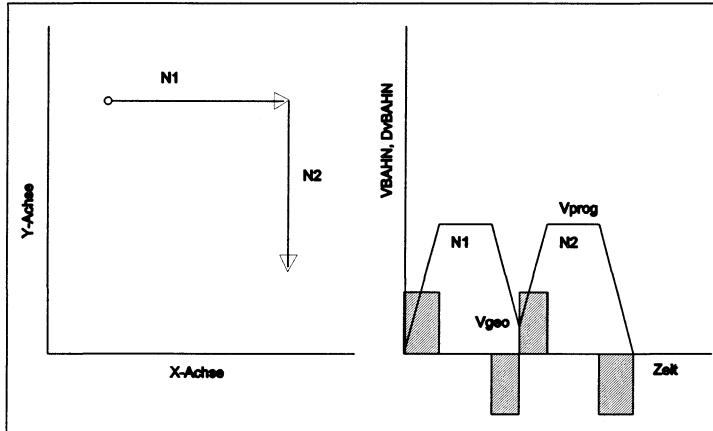
The circular radius error will be smaller than 9 [μm] at a larger circular radius of lower speed. The effective speed is reduced at a smaller circular radius.

The LAF function actually comprises 5 different functions.

- 1) Calculation of acceleration before interpolation
- 2) Recalculation and path profile
- 3) Milestone control
- 4) Path smoothing
- 5) Multiblock look ahead

1) Calculation of acceleration before interpolation

The LAF function serves to reduce the path speed at the discontinuities on the path so far that the drives are not at any time operated below their specified run-up values and the maximum acceleration torque is therefore not exceeded.

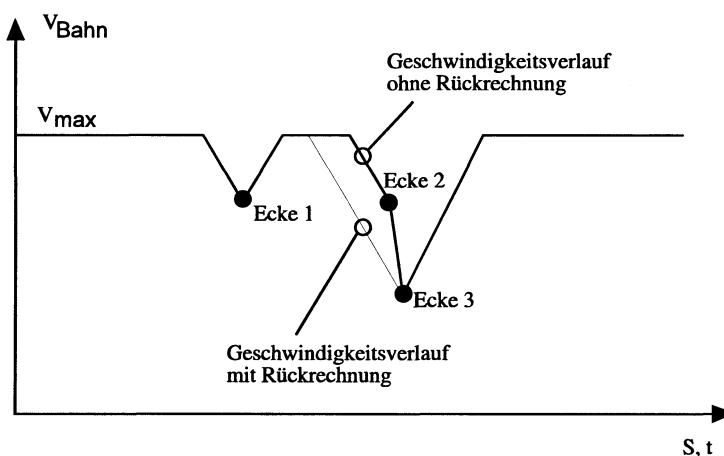


This figure shows the ideal path speed through a 90° corner. The shaded areas represent the path acceleration. In this example V_{geo} represents the minimum path speed which should be reached in order not to overload the drives.

The control options of the axes involved determine the maximum path speed (critical speed) at a 90° corner and the speed at which path acceleration this critical speed should be approached.

2) Recalculation and path profile

The LAF function not only locally "views" the path discontinuities (1 block look ahead feed), but it also determines in advance whether the established local critical speed is relevant at all between two blocks.



As shown in this figure, a critical speed has been established for corners 1, 2 and 3 by local "viewing" of the contour transitions. Since the critical speed of corner 3 can no longer be reached, when braking is not started until corner 2 has been approached, corner 2 should be considered irrelevant and corner 3 relevant. Relevance means that the path control has been informed of this corner before corner 2 is

reached.

This local critical speed may become lower:

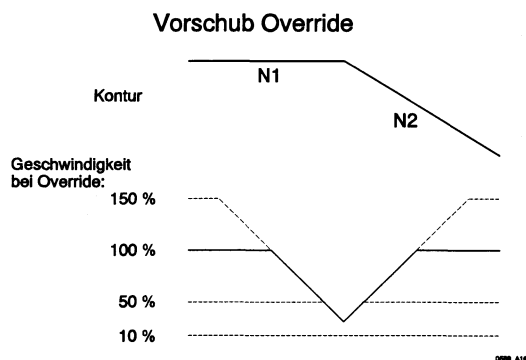
- Due to the approach path to this contour transition, the maximum achievable speed cannot be reached at all (braking should be possible in time).
- It may not be possible to reach this critical speed, because of a nearby lower critical speed point.

3) End point of synchronization

Because, as a rule, the axis-specific acceleration filter is not used in combination with the LAF functions, it must be ensured that at the path end the axes are braked at the specified positions and that these positions are not exceeded.

The maximum speed and acceleration values at the contour transition were established by the acceleration calculation. During the movement the path speed is limited by the axis-specific acceleration filter, if this is necessary because of the detectable braking distance. As a result, the maximum speed is lower than that of the normal look ahead.

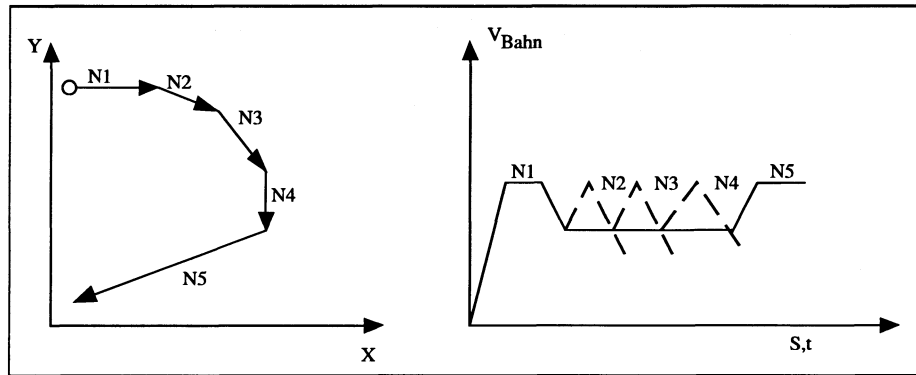
The calculated acceleration ramp does not depend on the actual override and is always optimal for the axes involved. If the override is set lower, the required braking distance is reduced accordingly. The braking ramp remains unchanged.



For override values > 100% the corner speed is, however, limited to the value calculated for 100%.

4) Path smoothing

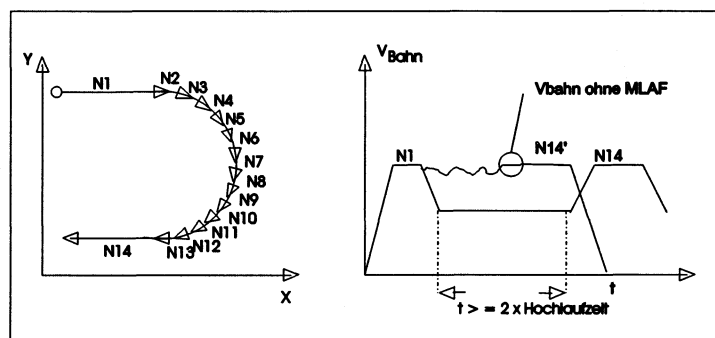
Especially when machining faces in the 2.5 and 3D ranges, you can put the LAF to optimal use thanks to complete checking of the path speed. For many path segments involving minor directional changes the maximum speed is slightly reduced. The path speed is then the same and not accelerated after every corner.



This figure shows the ideal path with and without smoothing. The path is not smoothed (dotted lines) when the LAF function generated single and valid minimum speeds.

5) Multiblock look ahead

Additionally to path smoothing, the local radius of curvature of the contour is calculated by approximation in the "extended" acceleration control, known as Multiblock look ahead. This is an addition to local "viewing" of the directional changes. This makes it possible to avoid too low a look ahead value by extremely short blocks with a directional change which is small individually, but large in total (circle).



Path with and without Multiblock look ahead. When Multiblock look ahead is activated, the path speed is reduced until the time required for making a semicircle is at least twice as long as the run-up time of the relevant axes.

84. F-Functions

Purpose

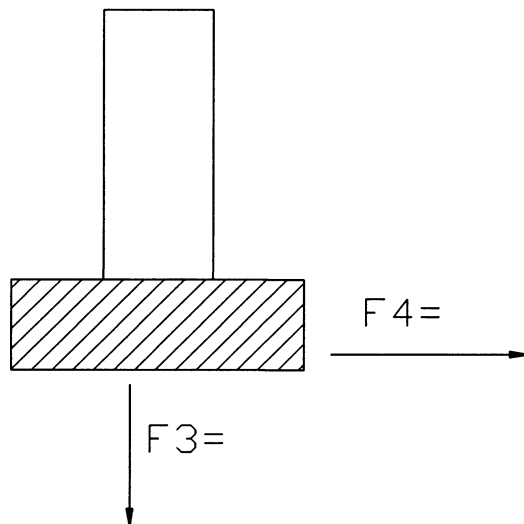
To set the feed in millimetres per minute or per revolution (mm/min or mm/rev). The feed rate actually used depends on several factors, for instance material, type of machining and tool.

Format

Up to V320: N... {F..} {F1=..} {F2=..}
 From V320: N... {F..} {F1=..} {F2=..} {F3=..} {F4=..}

Definitions, abbreviations:

F:	General feed for axis movements with G1/2/3
F1=:	Selection of constant cutting feed rate for radius compensation of circles.
F2=:	Retract feed at G85, infeed at G87/G89, G201 or measuring feed at G145.
F3=:	Feed for the (negative) infeed movement (infeed).
F4=:	Feed for the plane movement
Tool axis:	Axis perpendicular to the machining plane (G17, G18, ...).
Radial milling direction:	Milling in the machining plane
Axial milling direction:	Milling into the direction of the tool axis (in infeed direction only)



F4= Radial milling direction
 F3= Axial milling direction

Associated functions

G1, G2, G3, G41/G42, G45, G61/G62, G70/G71, G81/G89, G94/G95, G145, G201

Type of function

Modal	F, F1, F3, F4
Blockwise	F2

Notes and usage

The feed often has to be changed for technological reasons. The essential factors for making adjustments are:

- 1) Direction of the movement
- 2) Constant cutting feed rate for radius compensation of circles
- 3) Feed in cycles

1) Direction of the movement:

During cutting operations the feed should be carefully matched with the milling process for technological reasons. Technological conditions for milling in radial direction are different from those in axial direction.

It is most advantageous to the user if he is able to program 2 feed values modally and independently. Independent feed programming is possible with parameters F3= and F4=.

Feeds F, F3= and F4= are modal and programmable:

(0 ... 99999 [mm/min] metric)

(0 ... 9999.9 [inch/min] inch)

F3=: Sets the infeed

F4=: Sets the plane feed

F: Sets the infeed and plane feed

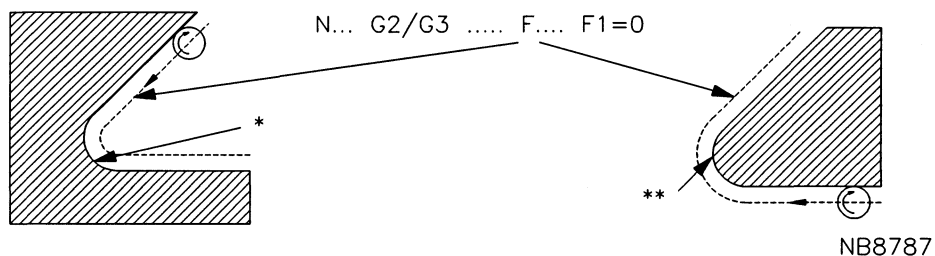
If F, F3= and F4= are programmed in a block, F3= and F4= have a higher priority than F.

2) Constant cutting feed

The parameter 'F1=' is used to ensure that the programmed feed rate along a workpiece contour remains constant regardless of the radius of the mill and the contour shape. This controlled velocity is called the CONSTANT CUTTING FEED.

F1=0 C.C.F. not applied (default mode; also set at CLEAR CONTROL or M30 or softkey CANCEL PROGRAM).

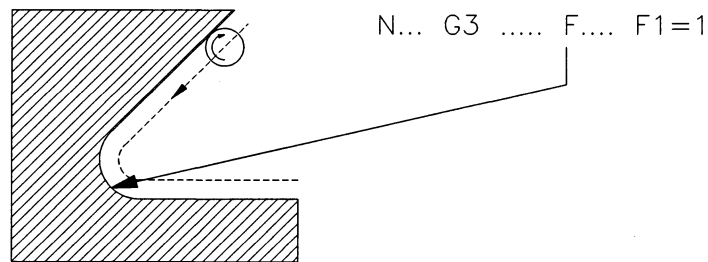
The programmed feed rate should be the velocity of the tool tip.



* Cutting feed too high

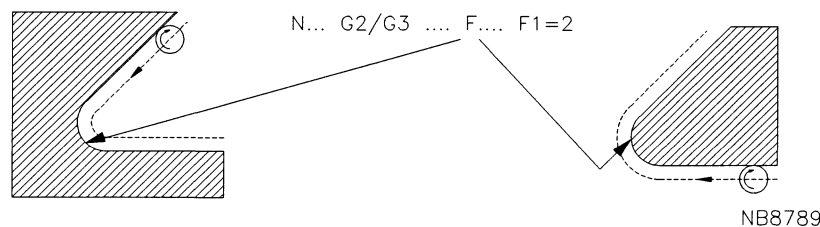
** Cutting feed too low

F1=1 C.C.F. applied only on the inside of arcs. The programmed feed rate is reduced to assure that the tool tip moves with the reduced velocity on the inside of an arc.



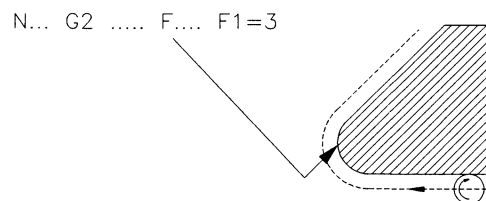
NB8788

F1=2 C.C.F. applied on the inside and outside of arcs. The programmed feed rate is reduced (inside are) or increased (outside are) to assure that the tool tip moves with the recalculated velocity. If the increased velocity is greater than the maximum feed rate (a Machine Constant value) the maximum feed rate is used.



NB8789

F1=3 C.C.F. applied only on the outside of arcs. The programmed feed rate is increased to assure that the tool tip moves with the increased velocity on the outside of an arc. If the increased velocity is greater than the maximum feed rate (a Machine Constant value) the maximum feed rate is used.



NB8790

- Infeed: only effective in the blocks dealing with infeed movements only.
- Plane feed: effective for all other movements not involving pure infeed movements.

Initialization: F3=0, F4=0 and F = 0

After M30, CANCEL PROGRAM Softkey or CLEAR CONTROL Softkey, F, F3= and F4= are zeroed.

3) Feed in cycles

In cycles G81, G83, G85 and G86 the movement in "axial" direction is not an infeed movement, but a feed movement. It is therefore programmed with F/F4=, not with infeed F3=.

In cycles G87, G88 and G89 the infeed movement can be programmed block by block using F2= and modally with F3=.

F3= is used as infeed in Easy Operate cycles.

Messages

- if feeds are missing (e.g. at F3=0 or F4=0 or F0).

Message: P04 No feed is programmed

MAXIMUM FEED

The maximum feed with which the machine tool may be operated is stated in the machine documents (MC740).

TECHNOLOGY TABLES

If the part program is entered or optimized through the control system, the spindle speed can be taken from the technology tables stored in the control system.

Material, machining type and tool should also be entered for making a selection from these tables.

Details of using the technology tables are given in the operating instructions.

Example

N10 F1000

N10 : Set feed to 1000 mm/min

85. Auxiliary function H

Purpose

Only the machine builder may use the H-function. Refer to the machine documentation.

Format

N... H...

Type of function

Depending on machine adaptation component.

Notes and usage

MAXIMUM NUMBER OF DECIMAL PLACES

The H-function can have a maximum of four decimal places.

86. Program stop M0

Purpose

To interrupt the execution of a program.

Format

N... {Programmed tool movement} M0

Associated functions

M1

Type of function

Non-modal

Notes and usage

ACTIVATION

The M0 function will become active when the current tool movement programmed in the same block has been executed.

SPINDLE SPEED AND COOLANT SUPPLY

The machine tool interface determines whether the spindle rotation and coolant are suppressed or cancelled as well.

Suppressed means that the spindle starts rotating and the coolant supply is switched on after resuming the execution of the part program.

Cancelled means that a spindle stop is executed and the coolant supply switched off. These functions have to be programmed again after the stop command.

RESUME PROGRAM EXECUTION

The execution of the program is resumed after the START button has been pressed.

Example

N200 G1 X100 Y100 F200 M0

N200: Move the tool to the programmed position and then halt the execution of the program.

Program stop M0

87. Optional program stop M1

Purpose

To interrupt the execution of a program if this function is encountered and the softkey OPTIONAL STOP in MACHINING is activated.

Format

N... {Programmed tool movement} M1

Associated functions

M0

Type of function

Non-modal

Notes and usage

ACTIVATION

The M1 function will become active when the current tool movement programmed in the same block has been executed.

SPINDLE SPEED AND COOLANT SUPPLY

The machine tool interface determines whether the spindle rotation and coolant are suppressed or cancelled as well.

Suppressed means that the spindle starts rotating and the coolant supply is switched on after resuming the execution of the program.

Cancelled means that a spindle stop is executed and the coolant supply switched off. These functions have to be programmed again after the stop command.

RESUME PROGRAM EXECUTION

The execution of the program is resumed after the start button has been pressed.

Note: If a M1, optional stop, is programmed and optional stop mode is not active then the following G1 blocks are executed with inpos instead of inpos disregarding the G28 function. In order to get correct behaviour the programmer should program a machine function in the block next to the M1-block.

Example

N200 G1 X100 Y100 F200 M1

N200: Move tool to programmed position and then halt program execution, if the softkey OPTIONAL STOP is active.

88. Spindle rotating clockwise/counter clockwise M3/M4

Purpose

To switch on spindle rotation in clockwise (CW) or counter clockwise (CCW) direction.

M3 Spindle rotation in clockwise direction.

M4 Spindle rotation in counter clockwise direction.

Format

N... {Programmed tool movement} M3/M4

Associated functions

M5, M19

M41, M42, M43, M44

Type of function

Modal

Notes and usage

START SPINDLE ROTATION

Spindle starts rotating before the tool movement programmed in the same block has been executed.

CANCELLATION

The direction of spindle rotation remains active until cancelled by:

- the opposite direction of rotation
- a SPINDLE STOP (M5 or M19)
- by END OF PROGRAM (M30) or CLEAR CONTROL.

PROGRAM STOP (M0/M1) OR TOOL CHANGE (M6/M66)

The machine tool interface determines whether the spindle rotation is suppressed or cancelled with a PROGRAM STOP or a TOOL CHANGE.

Suppressed means that the spindle starts rotating after resuming program execution.

Cancelled means that a spindle stop is executed and the direction of rotation has to be programmed again after a stop or tool change command.

Example

N20 G1 X100 Y100 S1000 M3

N20: Start spindle rotating in a clockwise direction at 1000 rev/min before starting tool movement to programmed position.

Spindle rotating clockwise/counter clockwise M3/M4

89. Spindle stop M5

Purpose

To stop spindle rotation. It depends on the interface of the machine tool if the coolant supply is switched off as well.

Format

N... {Programmed tool movement} M5

Associated functions

M3, M4, M19

M41, M42, M43, M44

Type of function

Modal

Notes and usage

ACTIVATION

The M5 function will become active when the current tool movement programmed in the same block has been executed.

The function remains active until a spindle rotation command is programmed.

Example

N35 G1 X50 Y50 F250 M5

N35: Execute tool movement and then switch off spindle rotation.

Spindle stop M5

90. Automatic tool change M6

Purpose

To interrupt program execution, retract the tool to a tool change position and perform a tool change, either automatically with a tool changer or manually.

Format

N... {T...} {T1=...} {T2=...} M6

Parameters

T Tool identification number.

T1= Activate/disable the cutting force monitor

T2= Use the extra tool offsets

Associated functions

M66, M67

Type of function

Non-modal

Notes and usage

START TOOL CHANGE

The tool change is executed, before the tool movement programmed in the same block has been executed.

Note:

In the following description the most commonly used sequence for the tool change command is given. Refer to the machine tool builder's documentation to see:

- if a movement to a tool change position is executed by the interface,
- which axes are involved,
- in which order the axes will move,
- if the spindle is stopped in an oriented position,
- if spindle speed and coolant supply are suppressed or cancelled.

MACHINE TOOL WITH AN AUTOMATIC TOOL CHANGER

The M6 function causes the following sequence to be performed:

- the tool first moves at rapid traverse rate to a tool change position.
- the old tool is then exchanged for a new tool and the new tools offsets made active.

RESUME PROGRAM EXECUTION AFTER THE AUTOMATIC TOOL CHANGE

The execution of the program continues automatically with the movement, if any, in the block with the tool change command.

MACHINE TOOL WITHOUT AN AUTOMATIC TOOL CHANGER

The M6 function causes the following sequence to be performed:

- the tool first moves at rapid traverse rate to a tool change position.
- the execution of the partprogram is halted, to allow the user to manually change the tool.

RESUME PROGRAM EXECUTION AFTER A MANUAL TOOL CHANGE

After the tool has been changed, the partprogram is restarted by pressing the START button. The movement, if any, in the block with the tool change command is executed.

SEARCHED TOOL (T)

If no T-word is programmed in a M6 block, the tool belonging to the last programmed tool number is loaded and its dimensions activated.

This situation occurs when the tool is searched during the execution of program blocks.

SPINDLE SPEED AND COOLANT SUPPLY

The machine tool interface determines whether the spindle rotation and coolant are suppressed or cancelled as well.

Suppressed means that the spindle starts rotating and the coolant supply is switched on after resuming the execution of the part program.

Cancelled means that a spindle stop is executed and the coolant supply switched off. These functions have to be programmed again after the tool change command.

TOOL CHANGE POSITION

It is advised to program all axes involved with the movement to a tool change position in the block with the tool change command or in the next block. In this way MANUAL BLOCK SEARCH and RESTART after program INTERRUPT is always executed in the same way.

INCREMENTAL PROGRAMMING AFTER A TOOL CHANGE

Increments with incremental programming are related to the last programmed position. A tool change position is not considered as a programmed position.

Example

N100: T12 M6

N100: Interrupt program execution to allow a new tool to be loaded. Tool no.12 offsets are made active.

91. Switch on coolant supplynr. 2 / Nr. 1. M7/M8

Purpose

To switch on coolant supply
M7: Nr. 2 coolant supply (internal coolant supply)
M8: Nr. 1 coolant supply

Format

N... {Programmed tool movement} M7/M8

Associated functions

M9, M13, M14

Type of function

Modal

Notes and usage

ACTIVATION

The coolant supply is switched on before the tool movement programmed in the same block, has been executed.

CANCELLATION

The function is active until cancelled by:

- COOLANT OFF (M9)
- END OF PROGRAM (M30) or CLEAR CONTROL.

SPINDLE STOP (M5)

It depends on the interface of the machine tool if the coolant supply is switched off with a SPINDLE STOP command.

PROGRAM STOP (M0/M1) OR TOOL CHANGE (M6/M66)

The machine tool interface determines whether the coolant supply is suppressed or cancelled with a PROGRAM STOP or a TOOL CHANGE.

Suppressed means that the coolant supply is switched on again after resuming program execution.

Cancelled means that the coolant supply is switched off and has to be programmed again after a stop or tool change command.

Example

N90: G1 X10 Y10 F200 M7

N90: Switch on No.2 coolant before executing programmed tool movement.

Switch on coolant supplynr. 2 / Nr. 1. M7/M8

92. Switch on coolant supplynr. Nr. 1. M8

Purpose

Switch on coolant No. 1 (in general external coolant supply).

Format

N... {Programmed tool movement} M8

Parameter

S	4	Spindle speed (U/min)
T	7.2	Toolnumber

Associated functions

-

Notes and usage

ACTIVATION -The M8 function is switched on before a tool movement programmed in the same program block is executed.

CANCELLATION

The function is active until cancelled by:

- COOLANT OFF (M9)
- END OF PROGRAM (M30) or CLEAR CONTROL.

Example

N90: G1 X15 Y15 F200 M8

N90: Switch on No.1 coolant before executing programmed tool movement.

Switch on coolant supplynr. Nr. 1. M8

93. Switch off coolant supply M9

Purpose

To switch off simultaneously the two coolant supplies.

Format

N... {Programmed tool movement} M9

Associated functions

M7, M8, M13, M14

Type of function

Modal

Notes and usage

ACTIVATION

The M9 function will become active when the current tool movement programmed in the same block has been executed.

The function remains active until one of the coolant functions (M7/M8, M13/M14) is activated again.

Example

N110: G1 X30 Y35 F150 M9

N110: Switch off coolant supplies after executing programmed tool movement.

Switch off coolant supply M9

94. Switch on nr. 1 Coolant and rotate spindle CW/CCW M13/M14**Purpose**

To switch on simultaneously the number 1 coolant supply and
 M13 spindle rotation in clockwise direction
 M14 spindle rotation in counter clockwise direction

Format

N... {Programmed tool movement} M13/M14

Associated functions

M3, M4, M5, M7, M8, M9
 M41, M42, M43, M44

Type of function

Modal

Notes and usage**ACTIVATION**

The spindle starts rotating and the coolant supply is switched on before the tool movement programmed in the same block, has been executed.

CANCELLATION OF ROTATING SPINDLE

The spindle remains rotating until cancelled by:

- a SPINDLE STOP (M5 or M19)
- END OF PROGRAM (M30) or CLEAR CONTROL.

CANCELLATION OF COOLANT SUPPLY

The coolant supply remains switched on until cancelled by:

- COOLANT OFF (M9)
- END OF PROGRAM (M30) or CLEAR CONTROL.

SPINDLE STOP (M5)

It depends on the interface of the machine tool if the coolant supply is switched off with a SPINDLE STOP command.

PROGRAM STOP (M0/M1) OR TOOL CHANGE (M6/M66)

The machine tool interface determines whether the spindle rotation and coolant are suppressed or cancelled with a PROGRAM STOP or a TOOL CHANGE.

Suppressed means that the spindle starts rotating and the coolant supply is switched on after resuming the execution of the part program.

Cancelled means that a spindle stop is executed and the coolant supply switched off. These functions have to be programmed again after the stop or tool change command.

Note:

A Machine Constant in the control must be set and the interface of the machine tool must be able to handle the functions M13 and M14. Refer to the machine tool builder's documentation to see if these functions can be programmed.

Example

N120 G1 X50 Y50 F100 S500 M13

Switch on no.1 coolant supply and rotate spindle clockwise at 500 rev/min before executing the programmed tool movement.

Switch on nr. 1 Coolant and rotate spindle CW/CCW M13/M14

95. Oriented spindle stop M19

Purpose

To stop the spindle in a fixed angular position, or, if an analogue spindle and a spindle transducer are used, in a programmed angular position.

Format

N... {D...} M19

Parameters

D Spindle offset value

Associated functions

M3, M4, M5, M13, M14

Type of function

Modal

Notes and usage

ANGULAR POSITION (D)

The angular position is measured from a fixed position, which is defined by a Machine Constant.

DIRECTION OF ROTATION

Moving the spindle to the measured position always occurs in a fixed direction defined by a Machine Constant.

SIGN OF D-WORD

D+: the angle in the defined direction of rotation

D-: the angle (360° - D-value) in the defined direction.

Note: The D-word is only available if a Machine Constant is set.

ACTIVATION

The M19 function will become active when the current tool movement programmed in the same block has been executed.

The spindle remains in its position, until a spindle rotation command is programmed or a M19 with another position.

Example

N125: D30 M19

N125: Stop the spindle +30° from the fixed angular position.

96. Measuring tool sizes M25

Purpose

To measure tool sizes, using a measuring probe with cube-shaped probe tip.

The G50 function is used to change the stored tool sizes, if the recorded sizes are beyond the specified range of limit values.

Format

N... G45 [I / J / K] X1=... M25

Parameter

I	+/-	2.3		Measuring direction for X axis
J	+/-	2.3		Measuring direction for Y axis
K	+/-	2.3		Measuring direction for Z axis
L	+/-	2.3		Measuring direction for B axis
X1=	+/-	6.3	5.4	Pre-measurement distance (mm inch)

Associated functions

-

Type of function

non modal

Notes and Usage

- G45 measuring cycle
- A measuring probe with cube-shaped probe tip is used for tool measurement. The probe is mounted in a fixed position.
 - The measuring position is loaded to the CNC memory.

MACHINE CONSTANT MEMORY

The following is stored in this memory:

- the coordinates of the fixed position of the measuring probe
- The sizes of the cube-shaped measuring probe tip.

Example

N90 G45 -I X1=5 M25

N90: Measure the tool in negative direction of the X-axis. The pre-measurement distance is 5 mm.

97. Calibrating the measuring probe M26

Purpose

The measuring probe radius is established by probing a calibration ring (ring gauge whose diameter is exactly known).

Format

Using the outer face of the ring gauge:

N... G46 [I+1,J+1 / J+1,K+1 / I+1,K+1] T... X1=... F... M26

Using the inner face of the ring gauge:

N... G46 [I-1,J-1 / J-1,K-1 / I-1,K-1] T... X1=... F... M26

I and J - gauge in XY plane; J and K - gauge in YZ plane; I and K - gauge in XZ plane

Parameters

I	+/-	1			Measuring direction for X axis (J or K should also be indicated)
J	+/-	1			Measuring direction for Y axis (I or K should also be indicated)
K	+/-	1			Measuring direction for Z axis (I or J should also be indicated)
F	+/-	4.3		3.4	Feed rate (mm/min inch/min)
T		7.2			Tool number
X1=	+/-	6.3		5.4	Pre-measurement distance (mm inch)

Associated functions

-

Type of function

non modal

Notes and usage

Measurement cycle	-	The G46/M26 measurement cycle is similar to the G46 measurement cycle. The difference between the measured centre point and the centre point stored in the machine constant memory is calculated. The F50 function is able to use this value for zero point shift.
-------------------	---	--

MACHINE CONSTANT MEMORY -
(MC242 / MC292 etc.).

The following is stored in this memory:

- the coordinates of the fixed position of the ring gauge
- the ring gauge diameter.

If X1= has not been programmed in the M26 block, a fixed preset value (machine constant) is used.

Example

N190 G46 I-1 K-1 T15 F50 M26:

Calibrate the probe by moving it to the outer face of the ring gauge in the XZ plane. The probe radius is stored in tool memory location T15. A fixed preset value (machine constant) is used for X1=.

98. Switch on / off a measuring probe M27/M28

Purpose

Before measurements can be performed with a remote signalling probe, e.g. an infrared probe, or a hard wired probe, the probe must be switched on and after using the probe, it must be switched off.

M27: switch on the measuring probe

M28: switch off the measuring probe

Format

Switch on a probe

N... M27

Switch off a probe

N... M28

Associated functions

G145

Type of function

Modal

Notes and usage

ACTIVATION

The function M27 is executed before the movements programmed in the same block, are executed and M28 after the movements in the block.

CANCELLATION

The measuring probe is switched off with the function M28, the function M30 (end of program) or at CLEAR CONTROL.

Switch on / off a measuring probe M27/M28

99. End of partprogram M30

Purpose

To terminate the execution of the partprogram and jump back to the begin of the program.

Format

N... M30

Type of function

Non-modal

Notes and usage

ACTIVATION

The M30 function will become active when the current tool movement programmed in the same block has been executed.

SPINDLE ROTATION AND COOLANT SUPPLY

When a M30 block is executed, the spindle is stopped and coolant supply switched off by the control.

DEFAULT SETTINGS

From G-functions belonging to one group, the default function of that group, if any, is automatically activated when the M30 function is executed.

Other functions with a default setting are reset too.

Example

```
N9001  
N1 ...  
:  
N... M30
```

Explanation

N9001:	Part program identification block.
N1 to N...:	Partprogram instructions.
N...	End of partprogram and jump back to begin of program

End of partprogram M30

100. Select spindle speed range M41/M42/M43/M44

Purpose

To select the gear range for the required spindle speed. In general the gear range is automatically selected by the control. In exceptional cases it may be necessary to change the speed without changing the gear range. This is done using the functions M41 - M44.

M41: first speed range
M42: second speed range
M43: third speed range
M44: fourth speed range

Format

N... S... M41/M42/M43/M44

Associated functions

M3, M4, M5, M13, M14, M19

Type of function

Modal

Notes and usage

SPEED RANGE SELECTION

The speed range can be selected:

- automatically by the CNC; the corresponding M-function is produced automatically by the CNC.
- by programming the corresponding M-function; useful when overlapping speed ranges are used.

SPEED RANGE LIMITS

The limits of the spindle speed ranges are stored in the MC-memory of the CNC.

TYPE OF SPEED RANGES

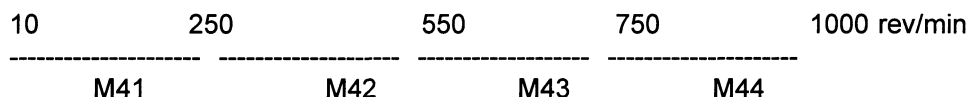
The spindle speeds can either be in separate speed ranges or in ranges which overlap each other.

If the M-function for range selection is not programmed and a programmed spindle speed occurs in two ranges, the highest range is automatically selected.

Example

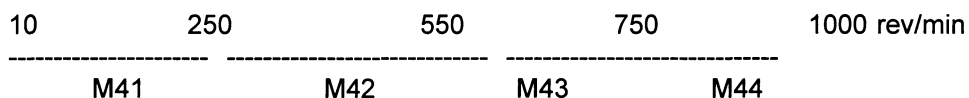
Example of speed ranges which do not overlap.

M41: 10 - 250 rev/min;
M42: 250 - 550 rev/min;
M43: 550 - 750 rev/min;
M44: 750 - 1000 rev/min.



Example of speed ranges which overlap.

M41: 10 - 250 rev/min;
M42: 200 - 550 rev/min;
M43: 500 - 750 rev/min;
M44: 700 - 1000 rev/min.



Programming example

N10: S50 M41

N10: Assumed are the speed ranges given above. A spindle speed of 50 rev/min is required. Therefore M41 is programmed because spindle range 1 has to be used. Automatic range selection is not being used.

101. Manual tool change M66

Purpose

To interrupt program execution without a retract to a tool change position, to allow a manual tool change to be performed.

Format

N... T... {T1=...} {T2=...} M66

Parameters

T Tool identification number.

T1= Activate/disable the cutting force monitor

T2= Use the extra tool offsets

Associated functions

M6, M67

Type of function

Non-modal

Notes and usage

APPLICATION OF M66

The function M66 is used with a tool:

- which is not in the tool magazine,
- for e.g. a back boring operation

START TOOL CHANGE

The tool change is executed, before the tool movement programmed in the same block has been executed.

MACHINE TOOL WITH AUTOMATIC TOOL CHANGER

The M66 function is used, when a tool is required which is not in the tool magazine. There is no automatic retraction to the tool change position and no execution of the tool change sequence.

Before performing manual tool change it might be necessary to unload the spindle (by programming T0 M6) and put the tool back in the tool magazine.

It might also be necessary to program a retract to a position where the tool can be loaded.

MACHINE TOOL WITHOUT AUTOMATIC TOOL CHANGER

The M66 function causes a halt in the program execution without a retraction to the tool change position, to allow the tool to be manually changed.

RESUME PROGRAM EXECUTION

After the tool change the program is restarted by pressing the START button. The movement, if any, in the block with the tool change command is executed.

SPINDLE SPEED AND COOLANT SUPPLY

The machine tool interface determines whether the spindle rotation and

coolant are suppressed or cancelled as well.

Suppressed means that the spindle starts rotating and the coolant supply is switched on after resuming the execution of the partprogram.

Cancelled means that a spindle stop is executed and the coolant supply switched off. These functions have to be programmed again after the tool change command.

Example

N200 T24 M66

N200: Interrupt program execution and change the tool manually. The tool dimensions of T24 become active.

102. Change tool compensation values M67

Purpose

To activate tool compensation values without a change of the physical tool being performed.

Format

N... T... {T2=...} M67

Parameters

T Tool identification number.

T2= Use the extra tool offsets

Associated functions

M6, M66

Type of function

Non-modal

Notes and usage

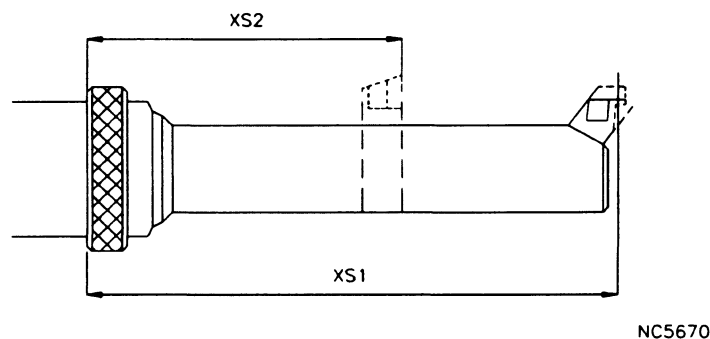
TOOLS WITH MORE THAN ONE CUTTING EDGE

When a tool with more than one cutting edge is used, eg. a boring bar, each cutting edge has its specific length and radius which are stored in the tool memory as offsets and extra tool offsets for the same tool.

ACTIVATE TOOL DIMENSIONS

The new tool dimensions are activated, before the tool movement programmed in the same block has been executed.

Example



The above boring bar, identified as tool T12, has two cutting edges. Edge 1 with tool length XS1 is stored in the tool memory with L=XS1 and edge 2 with tool length XS2 with L1 =XS2.

N100 T12 M6

N110 - N140

N150 T12 T2=1 M67

Explanation

N100: Load the boring bar; the offset values from T12 are used.

N110 - N140: Use cutting edge defined with T12

N150: A change of the offset values from XS1 to XS2. The boring bar itself is not changed.

103. Spindle speed S

Purpose

To set the speed in revolutions per minute (rev/min) of the main spindle of the machine tool. The speed that can be used, depends on a number of factors, such as material, operation, tool etc.

Format

N ... S....

Associated functions

M3, M4, M5, M13, M14, M19
M41, M42, M43, M44

Type of function

Modal

Notes and usage

MAXIMUM VALUE

A maximum speed of 50000 rev/min. can be programmed with the S-WORD.

Refer also to the documentation of the machine tool builder to see which spindle speeds can be used on the machine tool.

DIRECTION OF SPINDLE ROTATION

Refer to the description of M3/M4 for programming the direction of spindle rotation.

SPINDLE SPEED RANGES

Refer to the description of M41/M42/M43/M44 for selecting a spindle speed range.

TECHNOLOGY TABLES

If the part program is entered or updated via the control, it is possible to retrieve the spindle speed from the technology tables stored in the control.

The material, type of operation and tool have to be entered too to make the selection in these tables.

Refer to the User manual for details about using the technology tables.

Example

N10: S1000

N10: Set the spindle speed 1000 rev/min.

Spindle speed S

104. Tool number T

Purpose

To enter the tool identification number in the CNC Tool Memory. To indicate the tool to be loaded and call its offset values during the execution of the program. To initiate a search for another tool in the tool magazine.

Format

Automatic tool change

N... {T...} {T2=1/2} {T1=...} M6

Manual tool change

N... T... {T2=1/2} {T1=...} M66

Change offset values

N... T... {T2=1/2} M67

Tool change with sufficient tool life

N... T... T3=.. M6/M66

Parameters

T Tool identification number

T1= Activate/disable the cutting force monitor

T2= Use the extra tool offsets

Associated functions

M6, M66, M67

Type of function

Non-modal

Notes and usage

THE TOOL MEMORY

A special memory is available in the control to store the tool dimensions and other tool related parameters. Refer to the appendix TOOL MEMORY for a description of this memory.

MAXIMUM NUMBER OF TOOLS

A maximum of 255 sets of tool dimensions can be stored in the tool memory of the control. The actual number is stored as a machine constant.

TOOL IDENTIFICATION NUMBER (7)

The tool number in the tool memory is used to identify the tool. It is entered with the address T and a value with eight digits before and two digits behind the decimal point. The eight digits are reserved for the tool identification number. This number is also used in the part program and programmed with the T-words.

The two digits behind the decimal point specify a spare tool related to the tool.

If necessary these two digits can be used in the part program as well.

EXTRA TOOL OFFSETS (T2=)

A tool can have extra offsets. So the tool length, the tool radius and the corner radius of a mill can have three values: L, L1=, L2=, R, R1=, R2= and C, C1=, C2= respectively.

Which offsets are used when the tool is loaded, is indicated with the word T2= in the part program.

T2= not programmed: the offsets stated by L, R and C are used.

T2=1: the offsets stated by L1=, R1= and C1= are used. T2=2: the offsets stated by L2=, R2= and C2= are used.

Eg. a program block containing the words T1234 T2=2 M6 results in tool number 1234 being loaded together with its second offsets.

TOOL CHANGE WITH SUFFICIENT TOOL LIFE (T3=)

The T3= word in the part program indicates which replacement tool with sufficient residual tool life is used for a tool change.

A tool may have several replacement tools with different residual tool lives. (Refer to TOOL LIFE MONITORING in Appendix "TOOL MEMORY"). When T3= is programmed in a search block or during a tool change, a tool with sufficient residual tool life is searched.

Error P117 is displayed, if no tool with sufficient residual tool life is found.

Example:

A program block containing the words T1234 T3=1.1 M6 causes tool No. 1234 to be changed. The residual tool life is at least 1.1 minutes.

Note: If the life of a replacement tool expires during the machining operation, the program run is interrupted by error message P118. The program run will be resumed when the error message has been deleted and the Start key pressed. This error message is only activated if no other tools are changed during the operating sequence of Tool selection, Tool life run, Replacement tool selection and Replacement tool life run. To avoid this fault condition, the programmer may program an empty block preceding the tool change block, or he can avoid the situation described.

Note: If replacement tools outside the tool magazine range have been stored in the tool memory (location > MC28), error message P117 may be activated if a replacement tool is selected during the program run. To avoid this, the replacement tools in the tool magazine range should be stored in the tool memory. This applies, at the very least, to the replacement tools that may be used during the program run. If several replacement tools are envisaged for a particular tool in the tool memory, the tools with the lowest number of allocated replacement tools should be stored in the tool magazine range.

TOOL CHANGE COMMANDS (M6,M66,M67)

A physical tool change is commanded by one of the functions M6 (automatic tool change) or M66 (manual tool change) and a change of tool offsets by M67. Refer to these functions for details how to use them.

UNLOADING THE SPINDLE

With T0 M6 the spindle is unloaded and the tool put back at the position it originally left.

Unloading the spindle is necessary:

- before a manual tool change
- with oversized tools.

TOOL SEARCH

During the execution of a program, it is possible to search for the next tool in the magazine. So the tool is ready, when it should be loaded.

If the T-word is programmed without a tool change command, the search for the next tool is activated, provided that the interface of the machine tool allows a tool search.

SPARE TOOLS

A spare tool can replace the tool after its working life has ended or the lowest power level of the tool in the cutting force monitor is exceeded.

The spare tool is a two digit number placed behind the decimal point of the tool identification number.

SELECTING A SPARE TOOL

When a program block containing the words T1 M6 is executed, the CNC searches in the tool memory for the tool T1.xx.

If the tool with the lowest spare number has not been disabled, it is selected. If it cannot be used, another spare tool is selected which has a higher spare number.

If a tool identification and the spare tool number are programmed, eg. T1.05, spare tool 05 of tool 1 is used.

TOOL LIFE MONITORING

A working tool life is assigned to a tool. Every time the tool is used, the tool life is reduced with the cutting time. When the tool life has expired, a warning message is displayed, so that the tool can be replaced.

Note:

Refer to the appendix TOOL MEMORY for a description of the tool life monitoring and the actions taken when the tool life is exceeded.

TOOL BREAKAGE MONITOR

With an external device mounted on the machine tool, the tool length is measured when the tool is loaded into the spindle and again when it is put back into the magazine. If the difference between the two measurements is greater than a tolerance value, an error message is displayed and the tool disabled.

Note:

Refer to the appendix TOOL MEMORY for a description of the tool breakage monitoring and the actions taken when the tolerance value is exceeded.

CUTTING FORCE MONITORING

With an external device mounted on the machine tool, the cutting force being applied to a tool, can be monitored by constantly measuring the

power consumption of the spindle drive. When a power overload condition is detected, appropriate actions will be taken to prevent the workpiece or tool from being damaged.

ACTIVATING THE CUTTING FORCE MONITOR (T1=)

The cutting force monitor is activated with the word T1= and a value from 1 to 99 indicating the group of power levels to be used. The T1= word is programmed in a block containing one of the tool change functions (M6/M66).

DISABLING THE CUTTING FORCE MONITOR

The cutting force monitor is disabled by programming T1=0 or by not programming the T1= word.

Notes

1. Cutting force monitoring is usually used with heavy cutting, generally with tools of >10 mm diameter.
2. Refer to the appendix TOOL MEMORY for a description of the power levels with a cutting force monitor and the actions taken when a power level is exceeded.

105. E-parameters

Purpose

E-parameters are useful because they allow a more flexible use of programs: one program can be used for producing different workpieces by changing the parameter values stored in the CNC's Parameter Memory.

With the aid of macros and E-parameters a problem can be solved in general terms, eg. measuring a round hole in three or four points. At execution time the parameters receive their actual values and so the macro is adapted to the specific requirements of the program, eg. measuring a hole in three points.

Format

Parameter definition

N... E..=[Value or arithmetical expression]

Parameter allocation

N... [Address]= {+/-} E...

Parameter allocation and calculation

N... [Address]=[arithmetical expression]

Associated functions

G14, G22, G29

Type of function

Modal

Notes and Usage

CANCELLATION

Parameter values are modal unless changed by:

- assignment of new values in a partprogram,
- input via the user's panel in the parameter memory,
- input from a data carrier in the parameter memory.

Parameter values are not removed by CLEAR CONTROL.

NUMBER OF PARAMETERS

A maximum of 400 parameter values can be stored: a Machine Constant can be used for reducing this number.

ADDRESS

Any of the addresses available, except the address 'N'.

PARAMETER NUMBER (E)

This number specifies where the numerical value is stored in the CNC's Parameter Memory.

USE OF A PARAMETER IN MORE THAN ONE PROGRAM

A parameter can be used by different programs. If a new program uses a parameter which already has been assigned a value by a previous program, a new value must be assigned to the parameter otherwise the old value will be used again.

When a parameter is programmed but no value for that parameter is present in the Parameter Memory, an error message is produced.

STANDARD TYPES OF PARAMETERS

Parameters can be used in every CNC PILOT system. If the option EXTENDED ARITHMETICAL OPERATIONS is not available, the type of the parameter can be:

- integer, no decimal point: E1=20
- fixed point value: E1=200.105

INPUT ACCURACY WITH THE STANDARD TYPES

The input accuracy of the standard parameter types is:

- integer: a 15 digit number
- fixed point: at least 6 decimals behind the decimal point
maximum 15 decimals behind the decimal point

DISPLAY OF PARAMETER VALUES

The parameters stored in the Parameter Memory can be displayed on the screen of the control.

The displayed values are rounded values with a restricted number of decimals. They are displayed either as a fixed point value or as a value in the so called scientific notation, thus with an exponent.

Though the stored value has the same or a greater accuracy as the displayed one, the stored value might be different.

Eg. stored in the memory: 99.99999999
displayed : 100

STANDARD ARITHMETICAL OPERATIONS

The four arithmetical operations of addition (+), subtraction (-), multiplication (*) and division (:) are available in every CNC system.

E1=E2:	set E1 value equal to E2 value.
E1=E2+E3:	add the E3 value to the E2 value and store the result in E1.
E1 =E2-E3:	subtract the E3 value from the E2 value and store the result in E1
E1 =E2*E3:	multiply the E2 value by the E3 value and store the result in E1
E1=E2:E3:	divide the E2 value by the E3 value and store the result in E1.

RESTRICTIONS

1. Arithmetical expressions must not contain spaces between characters.
Eg. E1 = E2 is not allowed. This should be E1=E2
2. Arithmetical operators must be between arithmetical values eg.
E1=E2 E3 is not allowed.
3. Consecutive arithmetical operators are also not allowed eg.
E1=E2*:E3, except in the case of E1=E2*-E3.

Only one arithmetical operation is allowed in an expression.

EXTENDED ARITHMETICAL OPERATIONS (SOFTWARE OPTION)**Format**

Arithmetical operations

Exponentiation	E1=E2"E3
Square root	E1 =sqrt(E2)
Absolute value	E1 =abs(E2)
Integer conversion	E1 =int(E2)
Value of Pi(=3.141592..) pi	

Trigonometrical and inverse trigonometrical functions

sine	E1=sin(E2)
cosine	E1 =cos(E2)
tangent	E1 =tan(E2)
arcsin	E1 =asin(E2)
arccos	E1 =acos(E2)
arctan	E1 =atan(E2)

Relational expressions

equal	=	E1=E2=E3
not equal	<>	E1=E2<>E3 true E1=1
greater	>	E1=E2>E3
greater or equal	>=	E1=E2>=E3 not true E1=0
less	<	E1=E2<E3
less or equal	<=	E1=E2<=E3
Parentheses		

(...(...(...(...))...))): maximum of four levels

Notes:

1. If the relational expression is true E1=1, if not true E1=0

This can be used in the function for conditional jump (G29)

2. On a data carrier the mentioned functions should be in lower case characters.
3. In the format description, the parameters E2 and E3 represent any parameter or expression.
4. Functions and arithmetical expressions can also be used without parameters, eg.
X= (10+ 12*sin (23)).
5. The E-parameter containing the result of a calculation or a mathematical function, has the required accuracy, but different decimals can be stored.
Eg. E1=99.9999999 and E1=100.0000001 are two values with the same accuracy, but different decimals.
Problems may arise when using the function "int" or a relational expression in which all digits are compared.

BLOCK LENGTH

The size of an expression is restricted to 40 characters.

A program block can contain a maximum of 255 characters. This restricts the number of expressions that can be in a program block.

CONVERTING CALCULATED VALUES TO PROGRAM WORDS

Parameter (or calculated) values are automatically 'rounded' and converted by the CNC to the fixed number of decimals belonging to the program word.

Eg. programming $E1=101.74e-3$ and $X=E1$ causes the value to be rounded, so that $X0.102$ is obtained. The value is reduced to three digits after the decimal point.

EXTENDED PARAMETER FORMATS

A parameter value may be input in one of three formats: integer, fixed point and floating point.

1. Integer value (no decimal point): $E1=20$
2. Fixed point value: $E1=200.105$
3. Floating point value: $E1=1.965e5$

A floating point value is comprised of a fixed point number (the mantissa) which is multiplied by an exponential value eg. $1.965e5$ means $1.965 \times (10^5)$, which is equal to 196500.

INPUT ACCURACY WITH THE EXTENDED FORMAT

The input accuracy of the parameter types is:

- integer: a 15 digit number
- fixed point: at least 6 decimals behind the decimal point maximum 15 decimals behind the decimal point
- floating point: The mantissa is programmed as a fixed point value; the exponent is an integer between -99 to +99.

EXPONENTIATION (RAISING TO A POWER)

$E1=E2^2$ or $E1=E2^{E3}$ (with $E3=2$)

Both of the above operations result in the $E1$ parameter being made equal to the square of the $E2$ value.

Exponentiation operations are performed in a fixed sequence. The exponentiation operation is performed first and then the effect of the sign is included.

For example, the equation of $E1=-3^2$ is evaluated by first performing exponentiation (3^2) and then including the effect of the sign resulting in a negative value (-9).

If a negative number has to be raised to a power, the number should be enclosed by parentheses eg. $E1=(-3)^{E3}$. Another method is to assign the negative number to a parameter and then perform the exponentiation operation on the parameter eg. $E2=-3$ and then $E1=E2^2$.

Not allowed exponentiation calculations are:

1. 0^0 ;
2. $E2^{E3}$, when $E2 < 0$ and $E3$ has a real value.

RECIPROCAL

The reciprocal of E2 can be calculated by $E1=1:E2$ or $E1=E2^{-1}$

QUADRATE

The quadrate of E2 can be calculated by $E1=E2 \cdot E2$ or $E1=E2^2$

SQUARE ROOTS

The square root of E2 can be calculated by:

$E1=\text{sqrt}(E2)$ or $E1=E2^{.5}$

$E1=\text{sqrt}(\dots)$: between parentheses an arithmetical expression is allowed eg. $E1=\text{sqrt}(E2^2+E3^4)$.

Parameter E2 must be positive or zero when a square root (sqrt) calculation is performed.

ABSOLUTE VALUES

When the absolute function is used a negative value becomes positive. Positive values remain unchanged.

$E1=\text{abs}(E2)$

INTEGER VALUES

When the integer function is used the value is truncated. So all digits behind the decimal point are ignored.

Eg. $E2=1.9761e1$, and $E1=\text{int}(E2)$, the E1 value is truncated to 19.

Note:

1. The E-parameter is stored with the highest accuracy, but the user should be aware that different digits can be stored.

Eg. $E1=99.9999999$ $E3=100.0000001$

$E2=\text{int}(E1)$ gives $E2=99$ $E2=\text{int}(E3)$ gives $E2=100$

Both parameters E1 and E3 are stored with the same accuracy, the display shows in both cases the value 100, but the result of the function "int" is different.

2. It is advised to add a small value, eg. the required accuracy of the calculations, to the parameter of which the integer value is to be taken.

Eg. So if $E1=99.9999999$ or $E1=100.0000001$, the expression $E2=\text{int}(E1+.0000001)$ gives $E2=100$ independent of the value from E1.

THE CONSTANT PI

The value of pi is stored in the control with an accuracy of 15 digits. At each place, where a value or E-parameter is allowed, the word pi can be used. This constant can be used eg. with the conversion of angles from radians to decimal degrees or vice versa.

ANGLE IN DECIMAL DEGREES

The default programming mode for an angle is in degrees and decimal parts of a degree. This value can be entered directly in the trigonometric functions, arithmetical or relational expressions.

Eg. $E1=\sin(44.209303)$

ANGLE IN RADIANS

Sometimes with calculations in which angles are involved, it is useful to express the angle in radians.

360° equal 2π radians.

Therefore an angle of 44.209303° is equal to 0.7715979 radians.

If with the trigonometric functions the angle is in radians, the word rad has to be added to the value, thus:

Eg. $E1 = \sin(.7715979\text{rad})$

ANGLE CONVERSIONS

Degrees, minutes and seconds to decimal degrees:

An angle of 44° 12' 33.5" is converted into a decimal equivalent as follows:

N... $E1 = 44 + 12:60 + 33.5:3600$.

This conversion produces a decimal degree value of $E1 = 44.209303$.

Decimal degrees to radians

An angle of 44.209303° is converted into radians as follows:

N... $E1 = ((44.209303:360) * 2\pi)$

This conversion produces an angle in radians of $E1 = .7715979$

Radians to decimal degrees

An angle of 0.771579 radians is converted into decimal degrees as follows:

N... $E1 = (0.771579 * 360):(2\pi)$.

This conversion produces a decimal degree value of 44.209303 .

TRIGONOMETRICAL FUNCTIONS

The following trigonometrical functions are available: sinus (sin), cosine (cos), tangent (tan), These are written as:

$E1 = \sin(E2)$ $E1 = \cos(E2)$ $E1 = \tan(E2)$

For example, the sine of the angle 44.209303° can be programmed in any of the following ways:

$E1 = \sin(44.209303)$

$E1 = \sin(0.7715979\text{rad})$

Notes:

1. Parameter E2 represents any arithmetical expression.
2. Odd multiples of 90° cannot be used with the tan function; if this occurs an error message is generated.
3. If the angle is in radians, the word rad has to be programmed with the trigonometric functions.

INVERSE TRIGONOMETRICAL FUNCTIONS

The following inverse functions of the trigonometrical functions are available:

arcsin (asin), arccos (acos), arctan (atan).

These are written as:

E1 =asin(E2) E1 =acos(E2) E1 =atan(E2)

Notes:

1. Parameter E2 represents any arithmetical expression.
2. The values of the inverse functions asin and acos should be between -1 and +1; atan can have any numerical value.
3. The angle produced by these functions, is in decimal degrees.
4. The angle produced by asin and atan will be between -90° and +90°.
5. The angle produced by acos will be between 0° and 180°.

RELATIONAL EXPRESSIONS

The purpose of a relational expression is to set an E-parameter value to 1 when some conditions are met. If these conditions are not met the value of the parameter is set to 0.

This parameter can be used to perform jumps in the program by means of the G29 function.

The following relations can be used:

equal	=	E1=E2=E3
not equal	<>	E1=E2<>E3
greater	>	E1=E2>E3
greater or equal	>=	E1=E2>=E3
less	<	E1=E2<E3
less or equal	<=	E1=E2<=E3

Example

N.. G29 E1=E2>E3 E1 N=400

This block means:

If parameter E2 is greater than E3, the relation is true and thus parameter E1 set equal 1. Parameter E1 is used in the G29-block as the jump condition. So if E2>E3 a jump to N400 is executed.

Note:

1. Parameters E2 and E3 represent any arithmetical expression.
2. To satisfy a relational expression all digits are compared and have to be the same. When parameter values are produced by calculations, this may cause difficulties. In this case limits have to be set and checks performed to ensure that the value is within these limits.

PRIORITIES OF EVALUATING ARITHMETICAL AND RELATIONAL EXPRESSIONS

Arithmetical and relational operations are performed by the CNC in following order:

1. Evaluate functions: sin, cos, tan, asin, acos, atan, sqrt, abs, int.
2. Calculate reciprocals ($\wedge -1$) or perform exponentiation (\wedge).
3. Multiply (*) or divide (:).
4. Add (+) or subtract (-).
5. Evaluate relational expressions (=, <>, >, >=, <, <=).

When a block contains operations which have the same priority, the operations are evaluated from the beginning of the block to the end.

The block $E1=3+7:2-4\wedge 2+5*6$ is evaluated in the following order:

1. $4\wedge 2 = 16$
2. $7:2 = 3.5$
3. $5*6 = 30$
4. $3+3.5 = 6.5$
5. $6.5 - 16 = -9.5$
6. $-9.5 + 30 = 20.5$

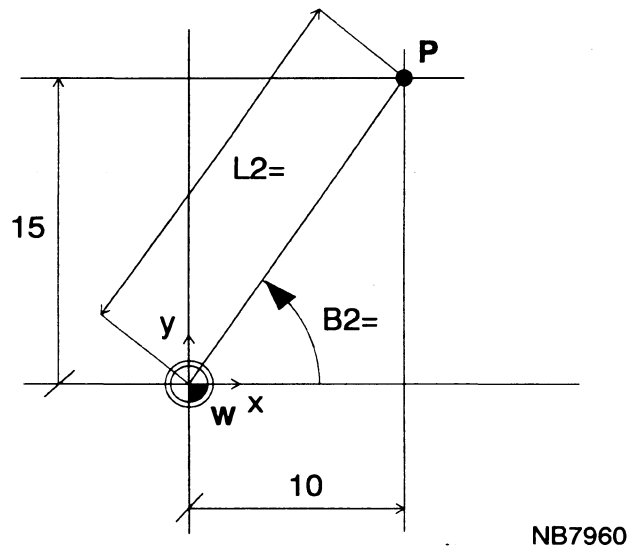
USE OF PARENTHESES ()

Parentheses () can be used to group operations and thus impose a different order of evaluating an expression. The expression between parentheses is evaluated in the normal sequence. Refer to PRIORITIES OF EVALUATING ARITHMETICAL AND RELATIONAL EXPRESSIONS for this sequence. After evaluating the expression the result is used.

A pair of parentheses can be placed within another pair; this is known as 'nesting'. The expression between each pair of parentheses is evaluated starting from the innermost 'nested' pair to the outermost pair. A maximum of four pairs of parentheses can be used in one expression.

Examples

EXAMPLE 1. Calculation of polar coordinates.



If the polar coordinates of point P related to the program datum W have to be calculated, the programming could be:

N100 B2=atan(15:10) L2=sqrt(10^2+15^2)

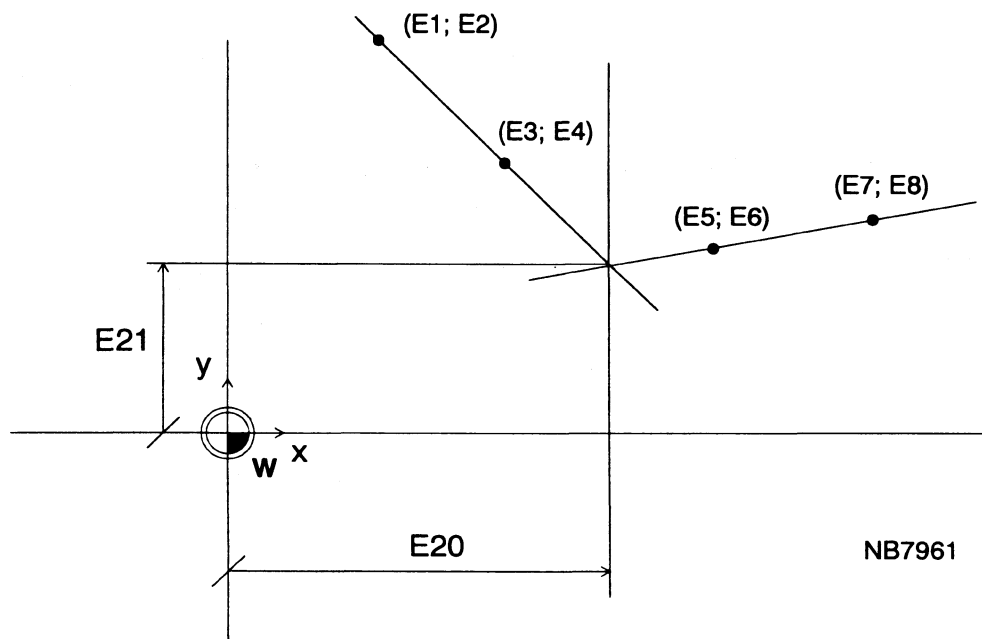
For B2= the calculation is performed in the sequence:

- calculate 15:10
- determine angle in decimal degrees

For L2= the calculation is performed in the sequence:

- calculate 10^2
- calculate 15^2
- add 10^2 and 15^2
- extract the square root.

EXAMPLE 2. Calculation of the intersection point of two lines.



Input parameters

- E1 : first coordinate of first point on first line.
 E2 : second coordinate of first point on first line.
 E3 : first coordinate of second point on first line.
 E4 : second coordinate of second point on first line.
 E5 : first coordinate of first point of second line.
 E6 : second coordinate of first point on second line.
 E7 : first coordinate of second point on second line.
 E8 : second coordinate of second point on second line.

Output parameters

- E20 : first coordinate of intersection point.
 E21 : second coordinate of intersection point.
 E79 =1: an error detected in the macro.
 =0: no error.

The macro

N99401(CALCULATE INTERSECTION POINT TWO LINES)

```

N1  E11=E3-E1 E12=E4-E2 E79=0
N2  E13=SQRT(E11^2+E12^2)
N3  E11=E11:E13 E12=E12:E13
N4  E13=E7-E5 E14=E8-E6
N5  E15=SQRT(E13^2+E14^2)
N6  E13=E13:E15 E14=E14:E15
N7  E16=E11*E13+E12*E14
N8  G29 E15=ABS(E16)<.99995 N=12 E15
N9  E79=1
N10 M0 (LINES ARE PARALLEL)
N11 G29 E79 N=17
N12 E15=E1-E7 E16=E2-E8
N13 E17=(E15*E12-E16*E11)
N14 E17=E17:(E13*E12-E11*E14)
N15 E20=E7+E17*E13
N16 E21 =E8+E17*E14
N17

```

Explanation:

N1-N3: the unit vector of the first line is calculated.
 N4-N6: the unit vector of the second line.
 N7-N8: check to see if the unit vectors are not parallel.
 N9-N11: if the lines are parallel, parameter E79 is set an error displayed with a program stop.
 After the start the calculations are not performed.
 N12-N14: compute the factor of the vector.
 N15-N16: calculate the coordinates of the intersection point.

Remark:

Parameter E79 can be used to handle the error in the activating program or macro.

Example of how the macro is used

First line through the points (30,50) and (60,30).
 Second line through the points (100,50) and (50,10).

The calculation of the intersection point could be programmed as:

```

N100 E1=30 E2=50      E3=60 E4=30
N101 E5=100           E6=50 E7=50 E8=10
N102 G22  N =99401
N103 G29  E79      K0 N=...
N104 G0    X=E20    Y=E21

```

Explanation:

N100: the points on the first line.
 N101: the points of the second line.
 N102: calculation of the intersection point.
 N103: if an error is detected, transfer control to block which contains M30.
 N104: move with rapid traverse to the intersection point.

106. Geometric calculations with continuous movements

CONVENTIONS WITH THE FORMATS

For all the formats in this appendix the G64-function is assumed to have already been programmed in a previous block and is therefore active.

The XY-plane is also assumed to be the active plane; if another plane is active the appropriate addresses must be substituted in the formats.

To show that more than one block is required with a particular format, the first block is numbered as N1 and the following as N2, N3 etc. The use of these numbers is not compulsory, they have been used purely as a convention.

In the formats, programming the end point is indicated with X.. Y... but instead of these coordinates the polar coordinates B2=, L2=.. or a defined point P.. can be used too.

Sometimes, programming the endpoint is indicated with [endpoint]. In this case the endpoint can be programmed as outside the geometry (G63 active). Thus with:
X.. only, or Y.. only, or X.. and Y.. or B1=.. and X.. or B1=.. and Y..

In the formats, programming the centre point of a circle is indicated with I.. J... but instead of these coordinates the polar coordinates B3=, L3=.. can be used.

The use of a support point is indicated with X.. Y.. I1=0 and of a parallel line with X.. Y.. I1= ... Instead of X.. and Y.. the polar coordinates B2=, L2=.. or a defined point P.. can be used too. It is also possible to use X1=.. Y1=.. instead of X.. Y.. I1=0 In some cases a support point or a parallel line can be used. This is indicated with X.. Y.. I1=0 or I1=+...

Sometimes a support point is indicated with [support point]. All possible formats for support point can be used.

In the illustrations in which the formats are explained, the following conventions are used:

P_o = a start point known from the previous block
P_s = a support point on a line or on a parallel line
P_e = a programmed end point
M = a programmed circle centre point
R = a programmed radius of a circle

A lot of line definitions are given with an angle B1=.. or a support point P, on the line. This is indicated in the illustrations with {B1=} and {Ps}. If B1= and Ps are drawn without the () both words have to be programmed.

CONTENTS OF FORMAT SECTION

INTERSECTION POINT

INTERSECTION POINT OF TWO STRAIGHT LINES

INTERSECTION POINT OF TWO LINES PROGRAMMED AS END POINT

CHAMFER BETWEEN INTERSECTING STRAIGHT LINES

ROUNDING BETWEEN INTERSECTING STRAIGHT LINES

ROUNDING BETWEEN STRAIGHT LINE AND CHAMFER

INTERSECTION POINT INDICATOR (J1=)

INTERSECTION POINT BETWEEN LINE AND CIRCLE

INTERSECTION POINT OF LINE AND CIRCLE PROGRAMMED AS END POINT

ROUNDING BETWEEN INTERSECTING LINE AND CIRCLE

INTERSECTION POINT BETWEEN CIRCLE AND LINE

INTERSECTION POINT OF CIRCLE AND LINE PROGRAMMED AS END POINT

ROUNDING BETWEEN INTERSECTING CIRCLE AND LINE

INTERSECTION POINT BETWEEN TWO CIRCLES

INTERSECTION POINT OF TWO CIRCLES PROGRAMMED AS END POINT

ROUNDING BETWEEN TWO INTERSECTING CIRCLES

POINT OF TANGENCY

POINT OF TANGENCY INDICATOR (R1=)

TANGENT LINE AND CIRCLE

CONNECTING CIRCLE BETWEEN TANGENT LINE AND CIRCLE

TANGENT CIRCLE AND LINE

CONNECTING CIRCLE BETWEEN TANGENT CIRCLE AND LINE

TWO TANGENT CIRCLES

CONNECTING CIRCLE BETWEEN TWO TANGENT CIRCLES

CONNECTING CIRCLE BETWEEN ELEMENTS WHICH DO NOT MEET

LINE AND CIRCLE

CIRCLE AND LINE

TWO CIRCLES OUTSIDE EACH OTHER

ONE CIRCLE INSIDE THE OTHER ONE

CONCENTRIC CIRCLES

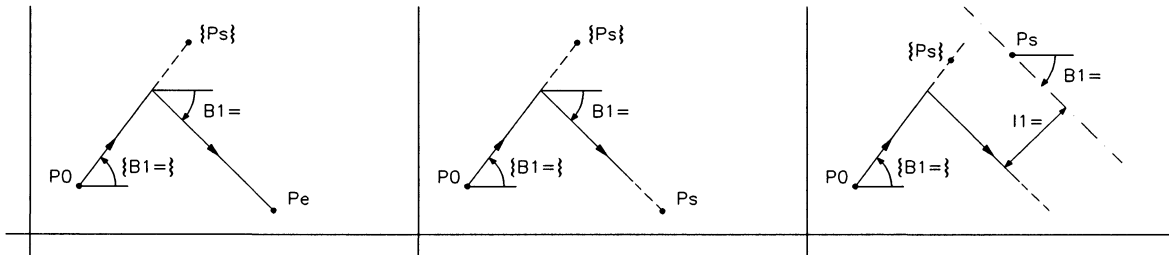
INTERSECTION POINT

INTERSECTION POINT OF TWO STRAIGHT LINES

To calculate the intersection point between two lines

start point from N1 is known

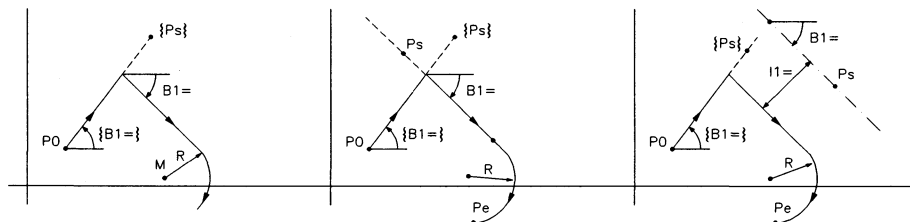
If the start point of the first line is known either the angle with the main axis or any support point on the line can be used to define the first line. Several formats are possible for the second line.



NB9665

```

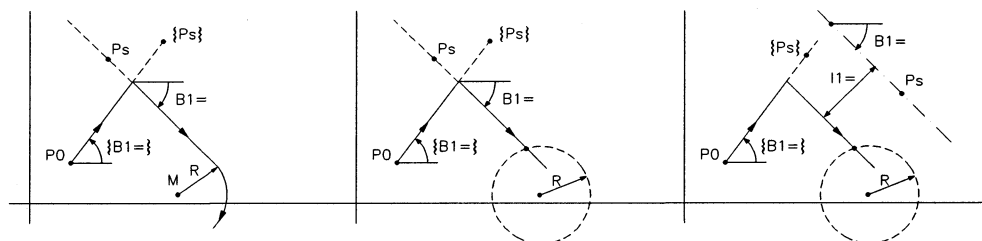
N1 G1 B1=..
N2   B1=.. X.. Y..
or
N1 G1 B1=..
N2   B1=.. X.. Y.. l1=0 or l1=±..
or
N1 G1 X.. Y.. l1=0
N2   B1=.. X.. Y..
or
N1 G1 X.. Y.. l1=0
N2   B1=.. X.. Y.. l1=0 or l1=±..
    
```



NB9668

```

N1 G1 B1=..
N2   B1=.. R1=0
or
N1 G1 B1=..
N2   B1=.. X.. Y.. l1=0 or l1=±.. R1=0
or
N1 G1 X.. Y.. l1=0
N2   B1=.. R1=0
or
N1 G1 X.. Y.. l1=0
N2   B1=.. X.. Y.. l1=0 R1=0
    
```



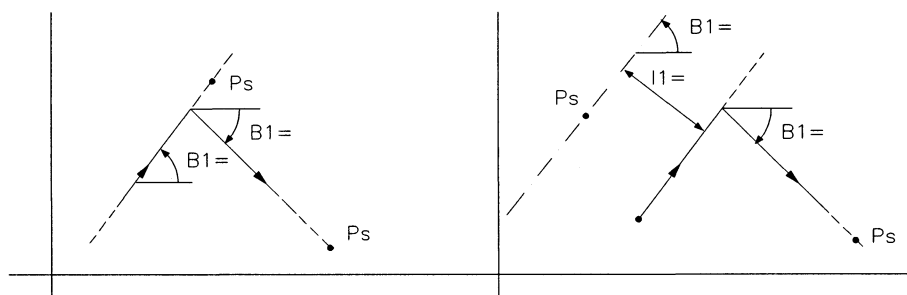
NB9669

```

N1 G1 B1=..
N2 X.. Y.. I1=0 R1=0
or
N1 G1 B1=..
N2 B1=.. X.. Y.. I1=0 or I1=.. J1=1/2
or
N1 G1 X.. Y.. I1=0
N2 X.. Y.. I1=0 R1=0
or
N1 G1 X.. Y.. I1=0
N2 B1=.. X.. Y.. I1=0 or I1=.. J1=1/2
    
```

start point from N1 is not known

If the start point from N1 is not known, both the angle and a support point have to be programmed in block N1. So this block reads:



NB9667

```

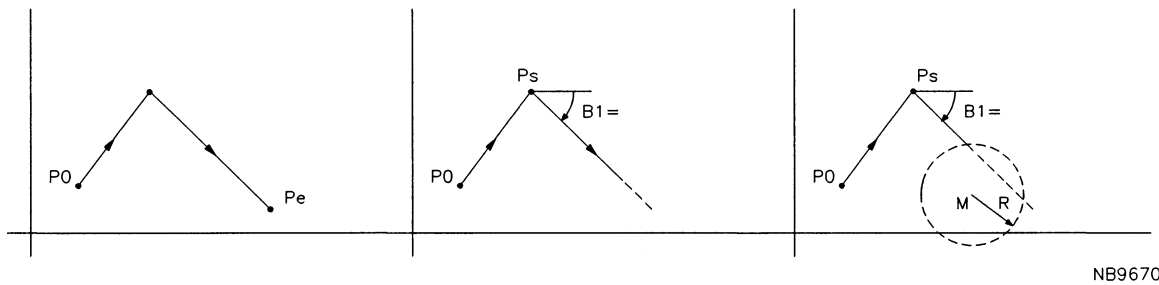
N1 G1 B1=.. X.. Y.. I1=0 or I1=..
    
```

Block N2 from the mentioned cases remains the same.

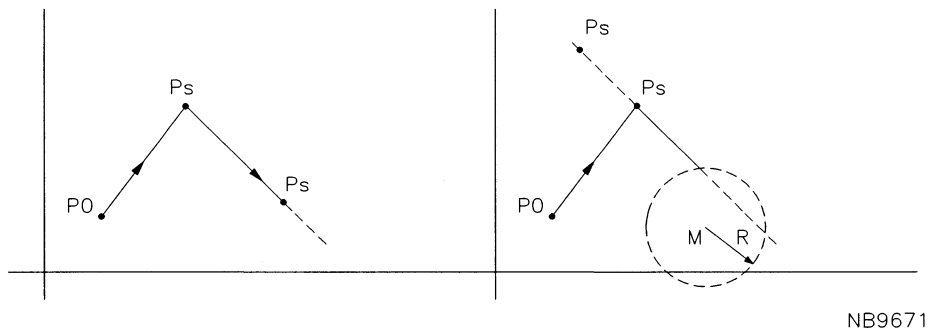
support point coincides with the point of intersection

If the support point coincides with the point of intersection, it is assumed that this point is the start point of the next line. This results in a few additional formats in which the second line can be programmed with either the angle or a support point on the line:

start point from N1 is known



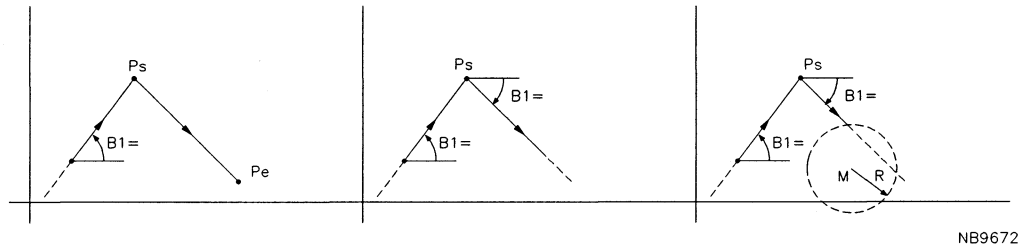
N1 G1 [support point=intersection point]
 N2 X.. Y..
 or
 N1 G1 [support point=intersection point]
 N2 B1=..
 or
 N1 G1 [support point=intersection point]
 N2 B1=.. J1=1/2



N1 G1 [support point=intersection point]
 N2 X.. Y.. I1=0
 or
 N1 G1 [support point=intersection point]
 N2 X.. Y.. I1=0 J1=1/2

start point from N1 is not known

If the start point from N1 is not known, both the angle and the support point have to be programmed in block N1. So this block reads:



NB9672

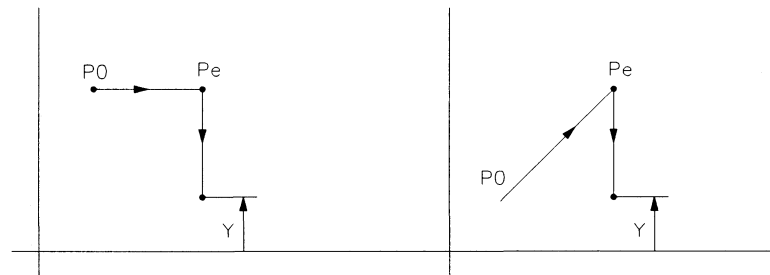
N1 G1 B1=.. [support point=intersection point]

Block N2 from the mentioned cases remains the same.

INTERSECTION POINT PROGRAMMED AS END POINT

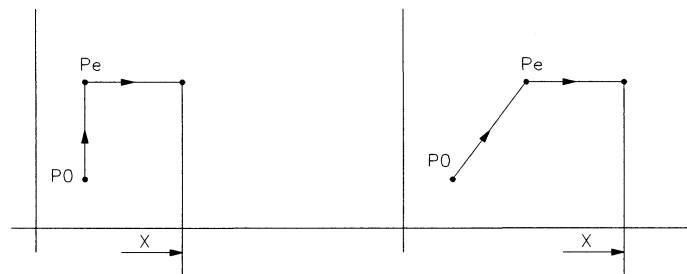
In some cases the intersection point of two lines is known from the drawing and can be programmed directly. It is assumed that this point is the start point of the next line. If the end point is programmed with one coordinate only, the other coordinate is picked up from the previous blocks. The following extra formats are possible:

start point from N1 is known



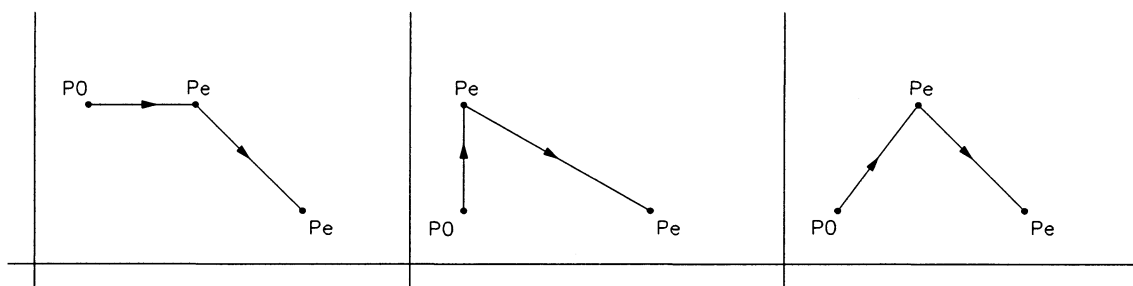
NB9674

N1 G1 X.. or X.. Y..
N2 Y..



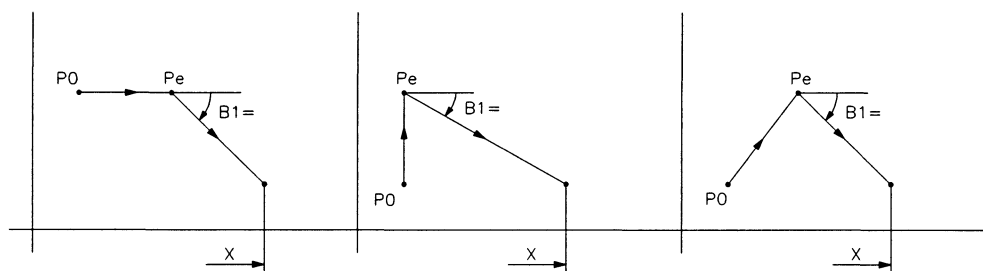
NB9675

N1 G1 Y.. or X.. Y..
N2 X..



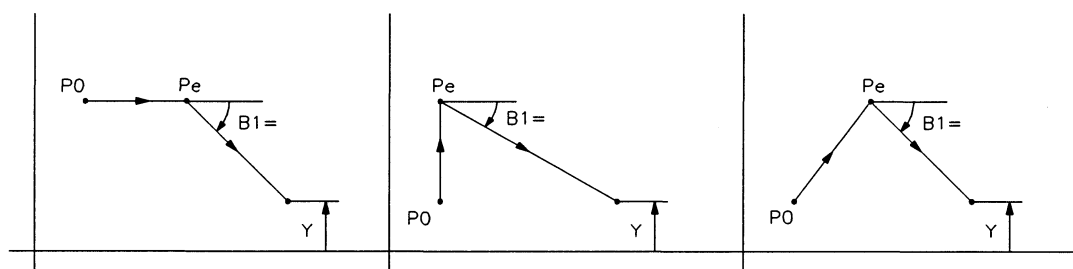
NB9676

N1 G1 X.. or Y.. or X.. Y..
N2 X.. Y..



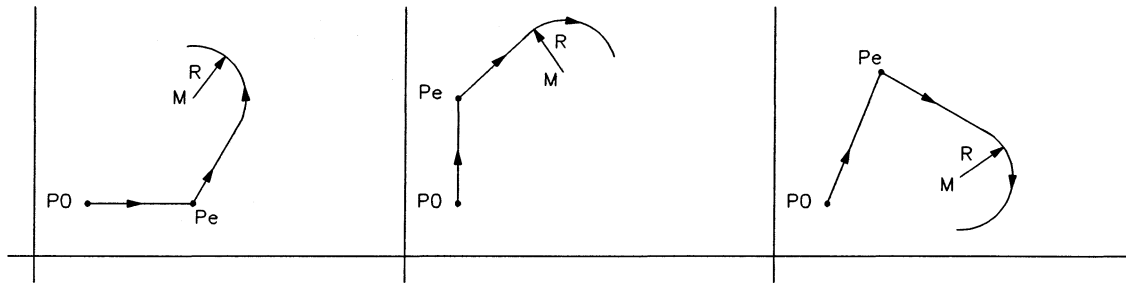
NB9677

N1 G1 X.. or Y.. or X.. Y..
N2 B1=.. X..



NB9678

N1 G1 X.. or Y.. or X.. Y..
N2 B1=.. Y..



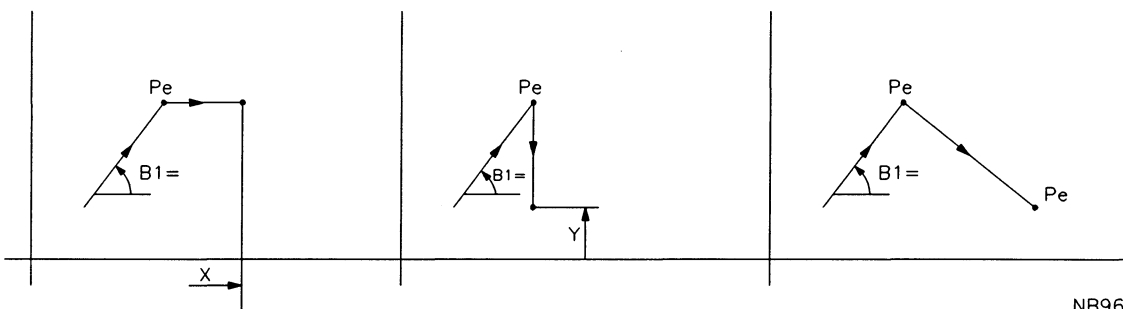
NB9679

N1 G1 X.. or Y.. or X.. Y..
N2 R1=0

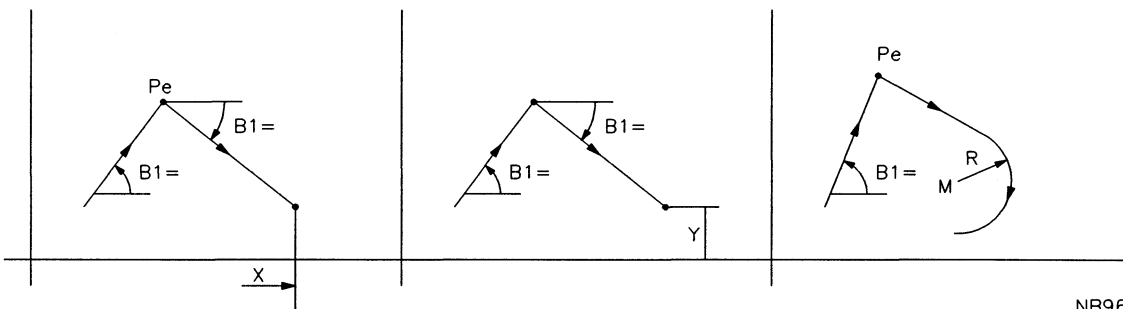
All formats from the previous case for calculating the intersection point can be programmed too.

start point from N1 is not known

If the start point from N1 is not known, both the angle and the endpoint have to be programmed in block N1. So this block reads:



NB9680



NB9681

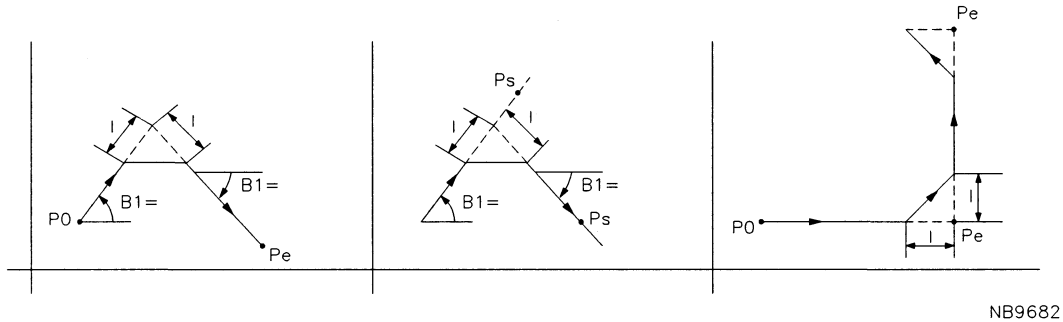
N1 G1 B1=.. X..
or
N1 G1 B1=.. Y..
or
N1 G1 B1=.. X.. Y..

Block N2 from the mentioned cases remains the same.

CHAMFER BETWEEN INTERSECTING STRAIGHT LINES

To insert a symmetrical chamfer between two straight lines

start point from N1 is known



N1 G1 B1=..
N2 l..
N3 etc.

Refer to block N2 of the previous sections for the formats of block N3.

Note: In stead of programming B1=.. in block N1 it is also possible to use a support point, a parallel line or an end point with either X.. or Y.. or X.. and Y..

start point from N1 is not known

If the start point from N1 is not known, both the angle and a support point or end point have to be programmed in block N1. So this block reads:

N1 G1 B1=.. X.. Y.. l1=0 or l1=..
N2 l..
N3 etc.

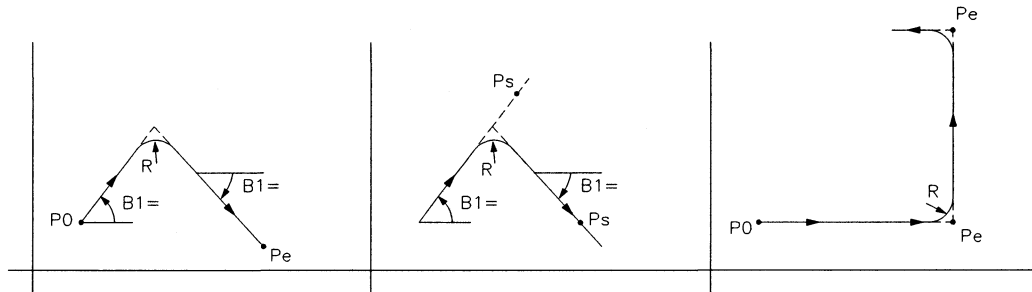
The blocks N2 and N3 are the same as with a known start point. So refer to that section for these blocks.

Note: In stead of programming B1=.. with a support point or parallel line in block N1 it is also possible to use an end point with either X.. or K. or X.. and Y..

ROUNDING BETWEEN INTERSECTING STRAIGHT LINES

To insert a rounding between two straight lines

start point from N1 is known



NB9683

N1 G1 B1=..
N2 G2/G3 R..
N3 G1 etc.

Note: In stead of programming B1=.. in block N1 it is also possible to use a support point, a parallel line or an end point with either X.. or Y.. or X.. and Y..

Refer to block N2 of the previous sections for the formats of block N3.

start point from N1 is not known

If the start point from N1 is not known, both the angle and a support point or end point have to be programmed in block N1. So this block reads:

N1 G1 B1=.. X.. Y.. I1=0 or I1=±..
N2 G2/G3 R..
N3 G1 etc.

Note: In stead of programming B1=.. with a support point or parallel line in block N1 it is also possible to use an end point with either X.. or Y.. or X.. and Y..

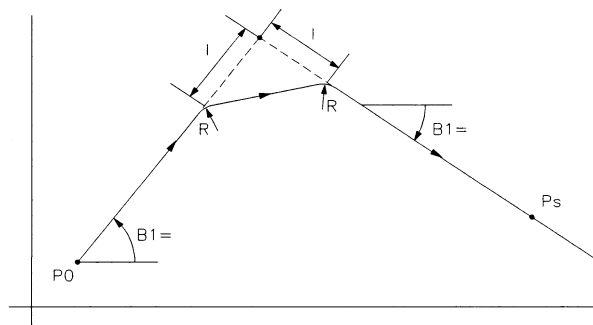
The blocks N2 and N3 are the same as with a known start point. So refer to that section for these blocks.

ROUNDING BETWEEN STRAIGHT LINE AND CHAMFER

To insert a rounding between a straight line and a chamfer

In the following formats both roundings are indicated. It is possible to insert just one rounding and leave out the other one.

start point from N1 is known



NB9684

```
N1 G1 B1=..
N2 G2/G3 R...
N3 G1 I...
N4 G2/G3 R...
N5 G1 etc.
```

Note: In stead of programming B1=.. in block N1 it is also possible to use a support point, a parallel line or an end point with either X.. or Y.. or X.. and Y..

Refer to block N2 of the previous sections for the formats of block N5.

start point from N1 is not known

If the start point from N1 is not known, both the angle and a support point or end point have to be programmed in block N1. So this block reads:

```
N1 G1 B1=.. X.. Y.. I1=0 or I1=±..
N2 G2/G3 R...
N3 G1 I...
N4 G2/G3 R...
N5 G1 etc.
```

Note: In stead of programming B1=.. with a support point or parallel line in block N1 it is also possible to use an end point with either X.. or Y.. or X.. and Y..

Refer to block N2 of the previous sections for the formats of block N5.

INTERSECTION POINT INDICATOR (J1 =)

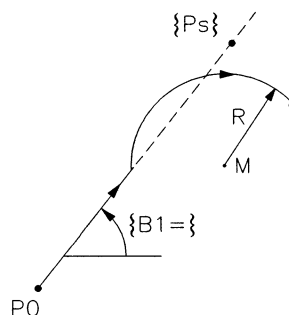
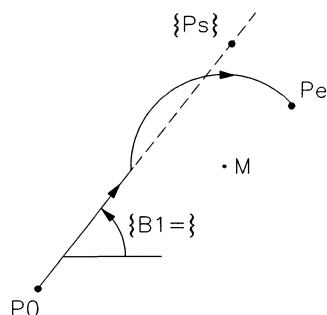
Refer to the description of the INTERSECTION POINT INDICATOR in the Notes and Usage of the function G64.

INTERSECTION POINT BETWEEN LINE AND CIRCLE

To calculate the point of intersection between line and circle

start point from N1 is known

If the start point of the line is known either the angle the line makes with the main axis or any support point on the line can be used to define the line. Several formats are possible for the circle:



NB9685

N1 G1 B1=.. J1=1/2

N2 G2/G3 I.. J.. X.. Y..

or

N1 G1 B1=.. J1=1/2

N2 G2/G3 I.. J.. R..

or

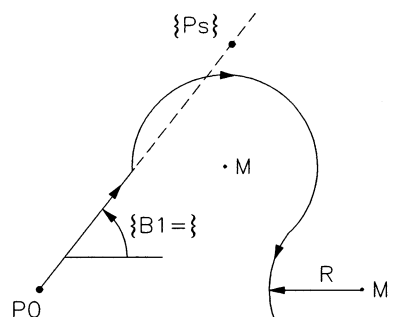
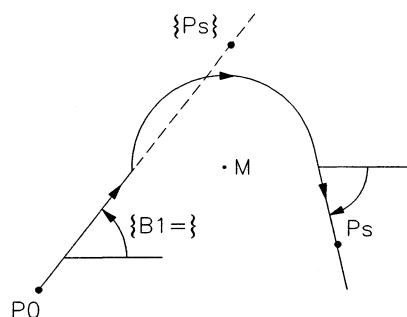
N1 G1 X.. Y.. I1=0 J1=1/2

N2 G2/G3 I.. J.. X.. Y..

or

N1 G1 X.. Y.. I1=0 J1=1/2

N2 G2/G3 I.. J.. R..



NB9686

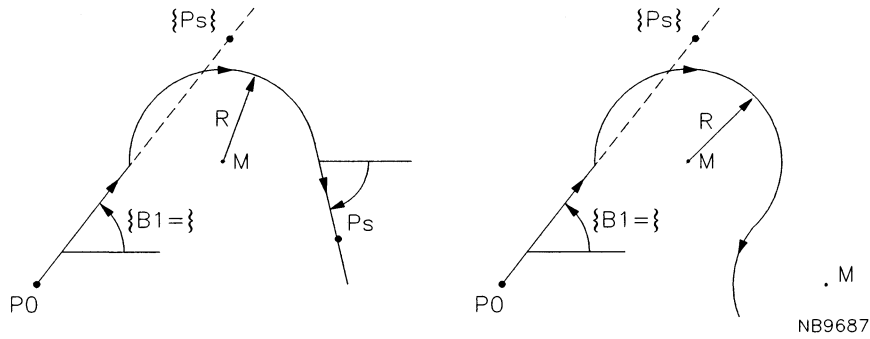
N1 G1 B1=.. J1=1/2

N2 G2/G3 I.. J.. R1=0

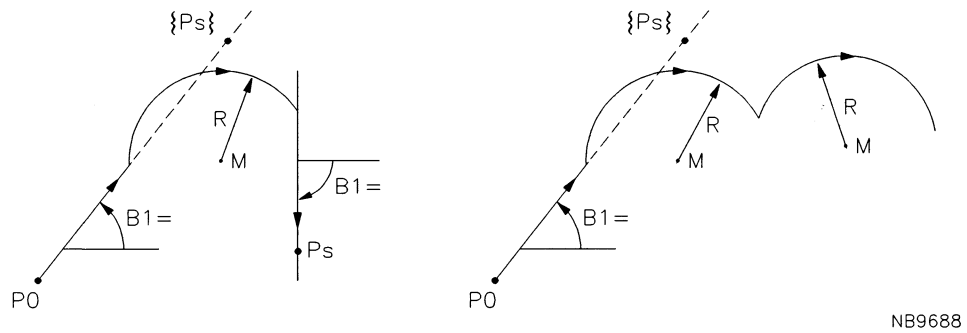
or

N1 G1 X.. Y.. I1=0 J1=1/2

N2 G2/G3 I.. J.. R1=0



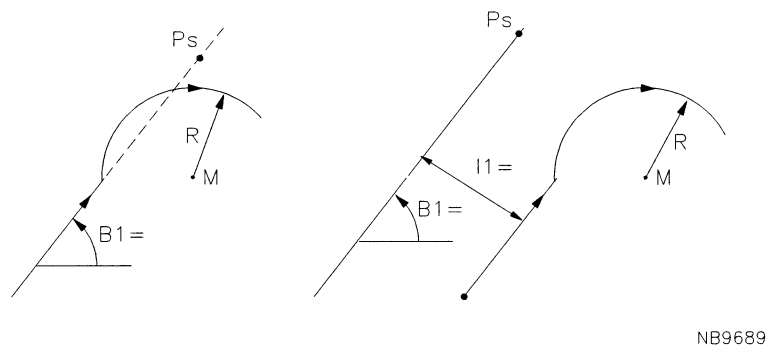
N1 G1 B1=.. J1=1/2
 N2 G2/G3 I.. J.. R.. R1=0
 or
 N1 G1 X.. Y.. I1=0 J1=1/2
 N2 G2/G3 I.. J.. R.. R1=0



N1 G1 B1=.. J1=1/2
 N2 G2/G3 I.. J.. R.. J1=1/2
 or
 N1 G1 X.. Y.. I1=0 J1=1/2
 N2 G2/G3 I.. J.. R.. J1=1/2

start point from N1 is not known

If the start point from N1 is not known, both the angle and a support point have to be programmed in block N1. So this block reads:



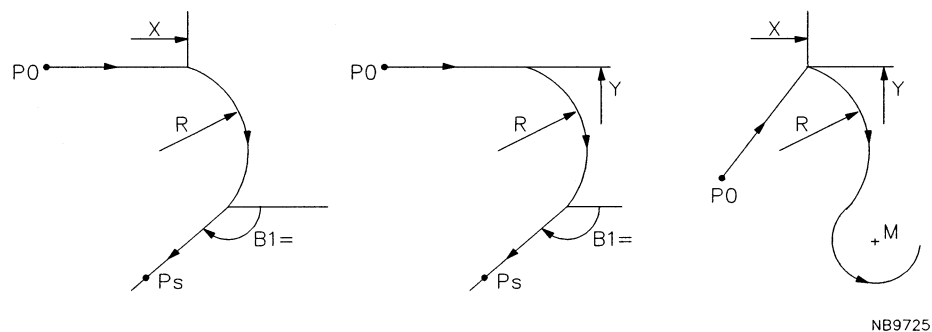
N1 G1 B1=.. X.. Y.. I1=0 or I1=.. J1=1/2

Block N2 from the mentioned cases remains the same.

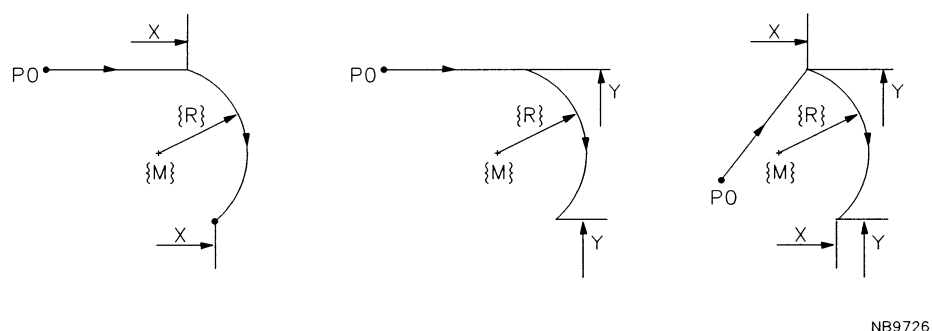
INTERSECTION POINT OF LINE AND CIRCLE PROGRAMMED AS END POINT

In some cases the intersection point of the line and circle is known from the drawing and can be programmed directly. It is assumed that this point is the start point of the next movement. The end point can be programmed with one or two coordinates and if the start point of the line is not known, the angle which the line makes with the main axis, can be added to the block. The following formats are possible:

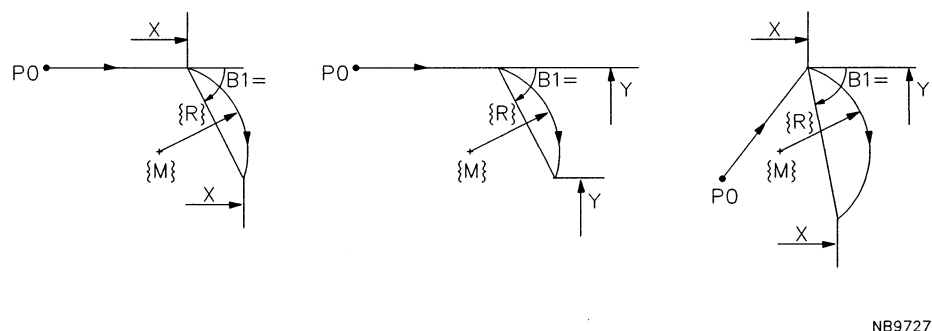
start point from N1 is known



N1 G1 X.. or Y.. or X.. Y..
N2 G2/G3 R.. R1=0

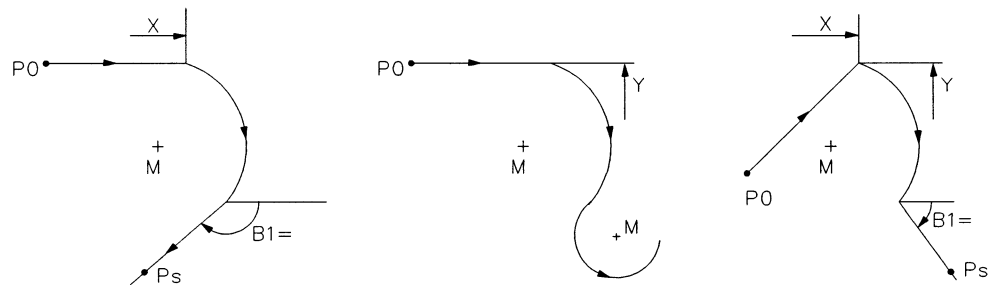


N1 G1 X.. or Y.. or X.. Y..
N2 G2/G3 R.. X.. or Y.. or X.. Y..
or
N1 G1 X.. or Y.. or X.. Y..
N2 G2/G3 I.. J.. X.. or Y.. or X.. Y..



N1 G1 X.. or Y.. or X.. Y..
N2 G2/G3 R.. B1=.. X.. or Y..

or
N1 G1 X.. or Y.. or X.. Y..
N2 G2/G3 I.. J.. B1=.. X.. or Y..

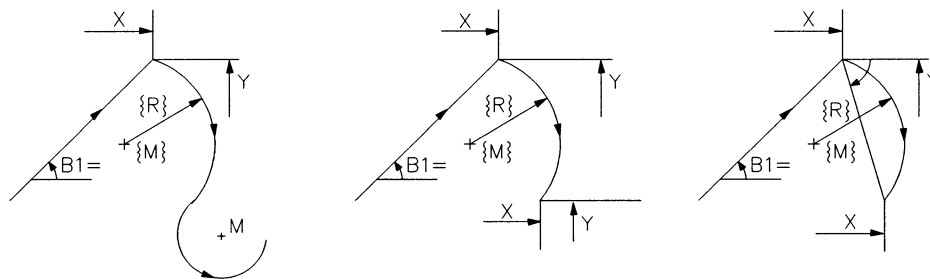


NB9728

N1 G1 X.. or Y.. or X.. Y..
N2 G2/G3 I.. J.. {R1=0} {J1=1/2}

start point from N1 is not known

If the start point from N1 is not known, the angle which the line makes with the main axis, has to be programmed in block N1 too. So this block reads:



NB9729

N1 G1 B1=.. X..
or
N1 G1 B1=.. Y..
or
N1 G1 B1=.. X.. Y..

Block N2 from the mentioned cases remains the same.

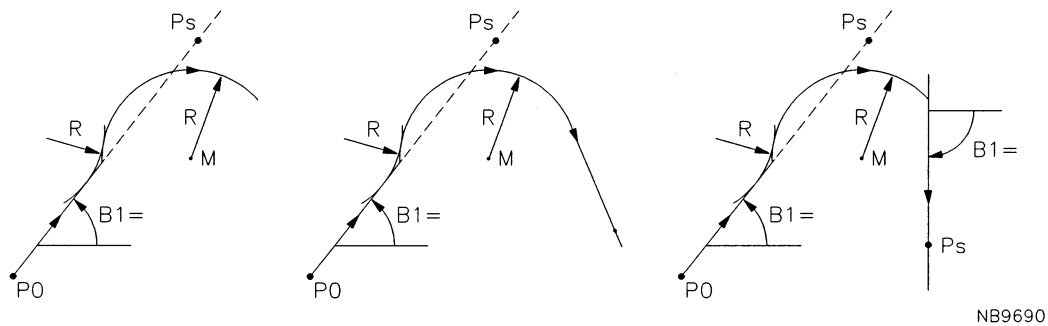
ROUNDING BETWEEN INTERSECTING LINE AND CIRCLE

To insert a rounding between an intersecting line and a circle

Notice that the direction of rotation of the rounding is opposite to that of the programmed circle.

calculated intersection point

start point from N1 is known



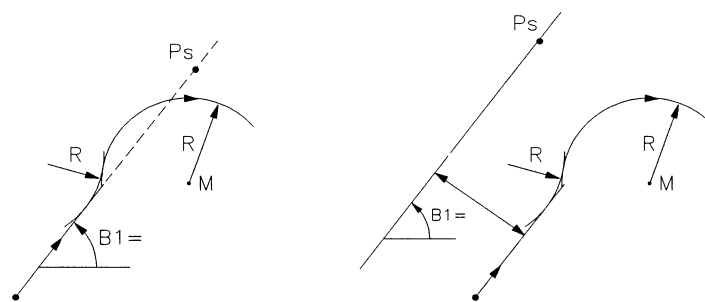
NB9690

N1 G1 B1=.. J1=1/2
 N2 G3/G2 R..
 N3 G2/G3 etc.
 or
 N1 G1 X.. Y.. I1=0 J1=1/2
 N2 G3/G2 R..
 N3 G2/G3 etc.

Refer to the section for calculating the intersection point for the formats of block N3.

start point from N1 is not known

If the start point from N1 is not known, both the angle and a support point have to be programmed in block N1, So this block reads:



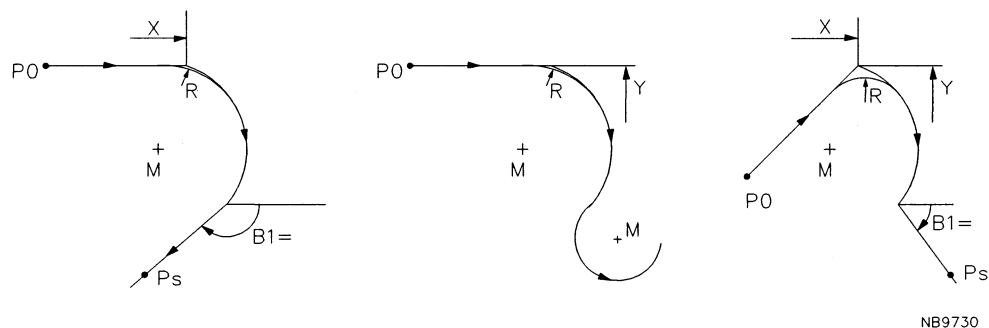
NB9691

N1 G1 B1=.. X.. Y.. I1=0 or I1=.. J1=1/2
 N2 G3/G2 R..
 N3 G2/G3 etc.

Refer to the section with the known start point for the formats of block N3.

programmed intersection point

start point from N1 is known



NB9730

N1 G1 X.. or Y.. or X.. Y..
 N2 G3/G2 R..
 N3 G2/G3 etc.

Refer to the section for programming the intersection point for the formats of block N3.

start point from N1 is not known

If the start point from N1 is not known, the angle which the line makes with the main axis, has to be programmed in block N1 too. So this block reads:

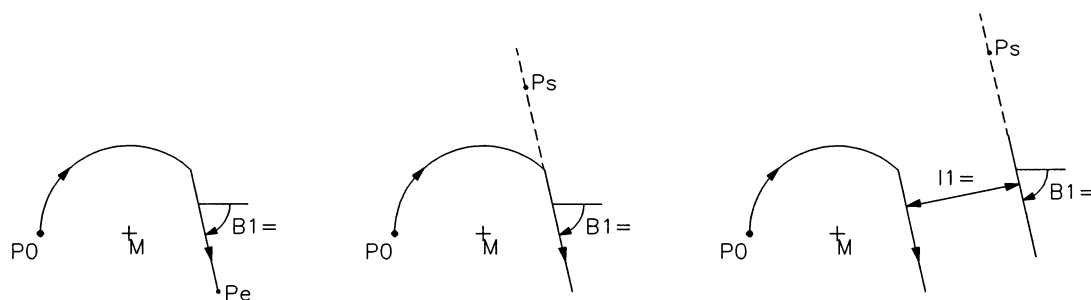
N1 G1 B1=.. X.. or Y.. or X.. Y..
 N2 G3/G2 R..
 N3 G2/G3 etc.

Refer to the section with the known start point for the formats of block N3.

INTERSECTION POINT BETWEEN CIRCLE AND LINE

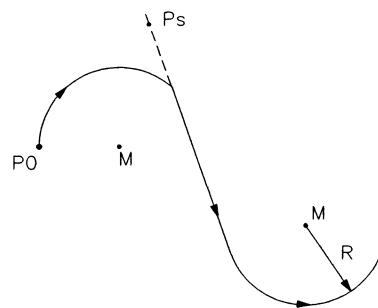
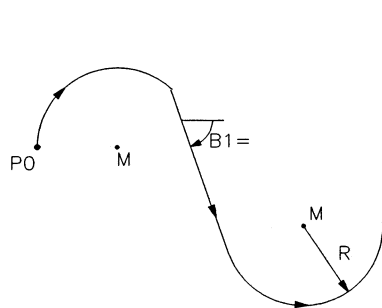
To calculate the intersection point between circle and line

start point from N1 is known



NB9692

N1 G2/G3 I.. J.. J1=1/2
 N2 G1 B1=.. X.. Y..
 or
 N1 G2/G3 I.. J.. J1=1/2
 N2 G1 B1=.. X.. Y.. I1=0 or I1=..

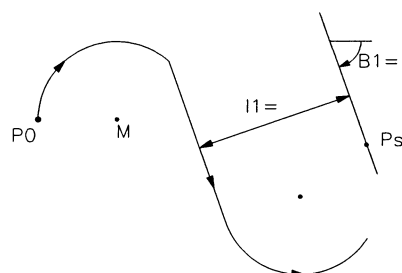
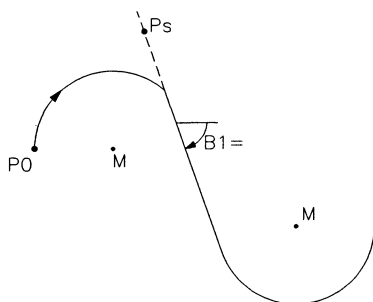


NB9693

N1 G2/G3 I.. J.. J1=1/2
N2 G1 B1=.. R1=0

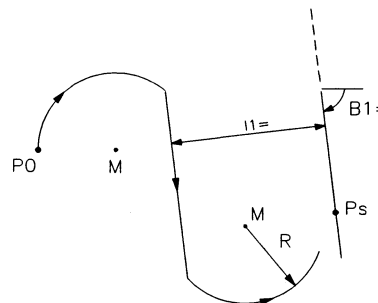
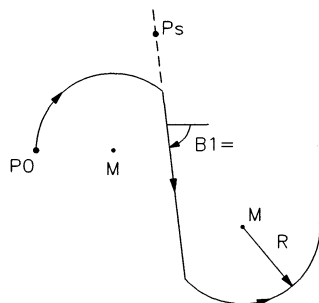
or

N1 G2/G3 I.. J.. J1=1/2
N2 G1 X.. Y.. I1=0 R1=0



NB9694

N1 G2/G3 I.. J.. J1=1/2
N2 G1 B1=.. X.. Y.. I1=0 or I1=.. R1=0

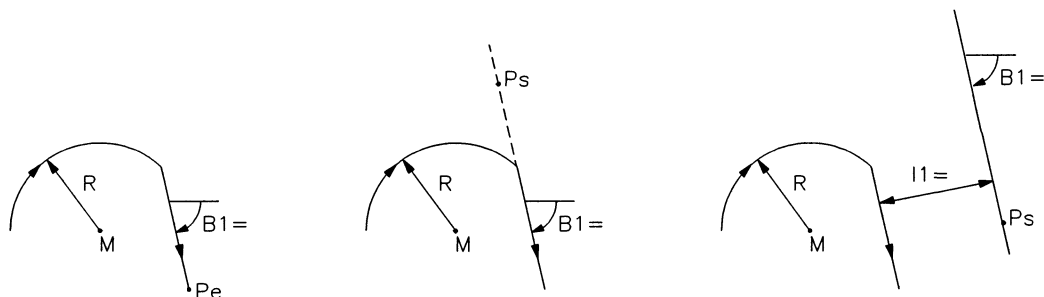


NB9695

N1 G2/G3 I.. J.. J1=1/2
N2 G1 B1=.. X.. Y.. I1=0 or I1=.. J1=1/2

start point from N1 is not known

If the start point is not known, the radius value (R-word) should also be programmed in the block N1, thus this block reads:



NB9696

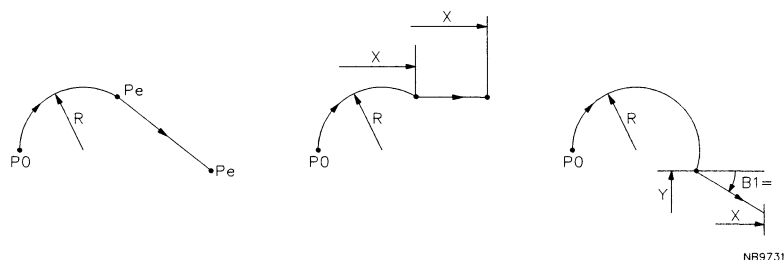
N1 G2/G3 I.. J.. R.. J1=1/2

Block N2 from the mentioned cases remains the same.

INTERSECTION POINT OF CIRCLE AND LINE PROGRAMMED AS END POINT

In some cases the intersection point of the circle and line is known from the drawing and can be programmed directly. It is assumed that this point is the start point of the next movement. The following formats are possible:

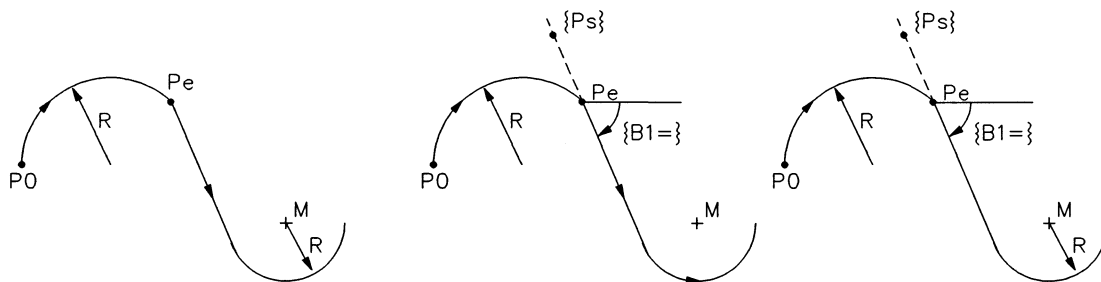
start point from N1 is known



NB9731

N1 G2/G3 R.. [endpoint]

N2 G1 [endpoint]

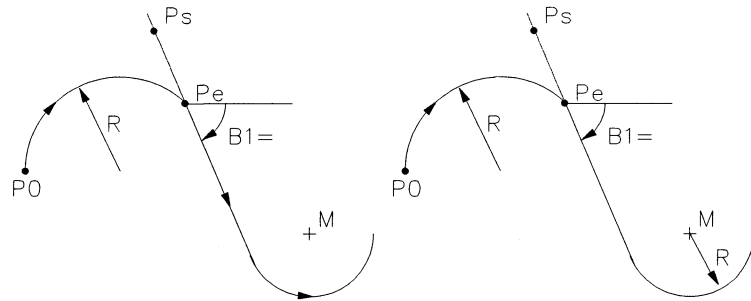


NB9732

N1 G2/G3 R.. [endpoint]

N2 G1 R1=0

or
 N1 G2/G3 R.. [endpoint]
 N2 G1 B1=.. {R1=0} {J1=1/2}
 or
 N1 G2/G3 R.. [endpoint]
 N2 G1 X.. Y.. I1=0 {R1=0} {J1=1/2}

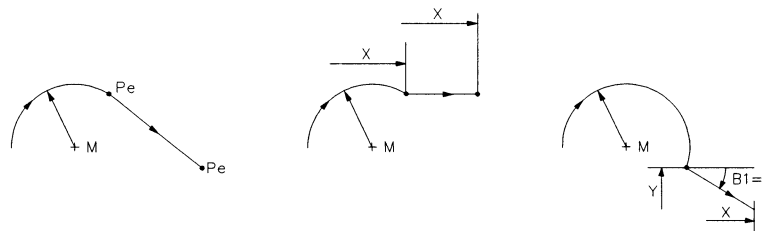


NB9733

N1 G2/G3 R.. [endpoint]
 N2 G1 B1=.. X.. Y.. I1=0 or I1=.. {R1=0} {J1=1/2}

start point from N1 is not known

If the start point from N1 is not known, the centre point coordinates instead of the radius have to be programmed in block N1. So this block reads:



NB9734

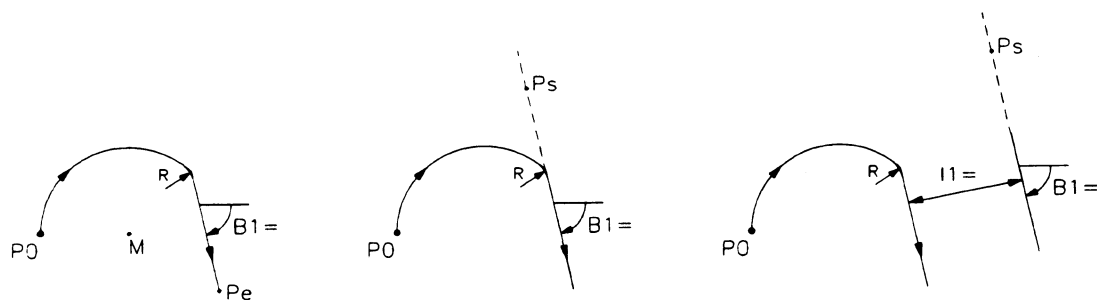
N1 G2/G3 I.. J.. [endpoint]

Block N2 from the mentioned cases remains the same.

ROUNDING BETWEEN INTERSECTING CIRCLE AND LINE

To insert a rounding between an intersecting circle and a line. Notice that the direction of rotation of the rounding is opposite to that of the programmed circle.
calculated intersection point

start point from N1 is known



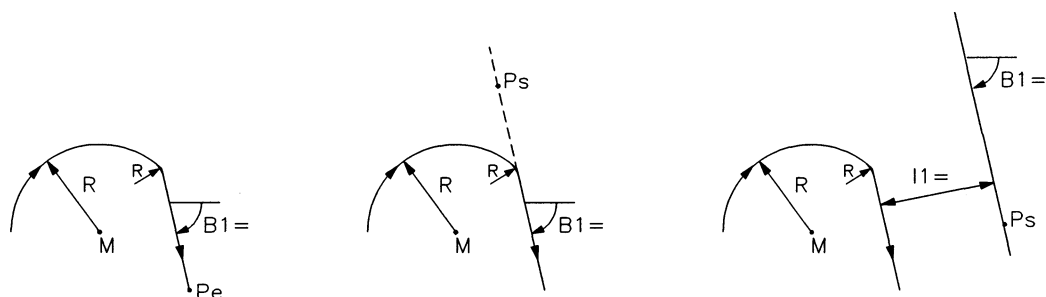
NB9697

N1 G2/G3 I.. J.. J1=1/2
N2 G3/G2 R..
N3 G1 etc.

Refer to the section for calculating the intersection point for the formats of block N3

start point from N1 is not known

If the start point is not known, the value (R-word) should also be programmed in the block N1, thus this block reads:



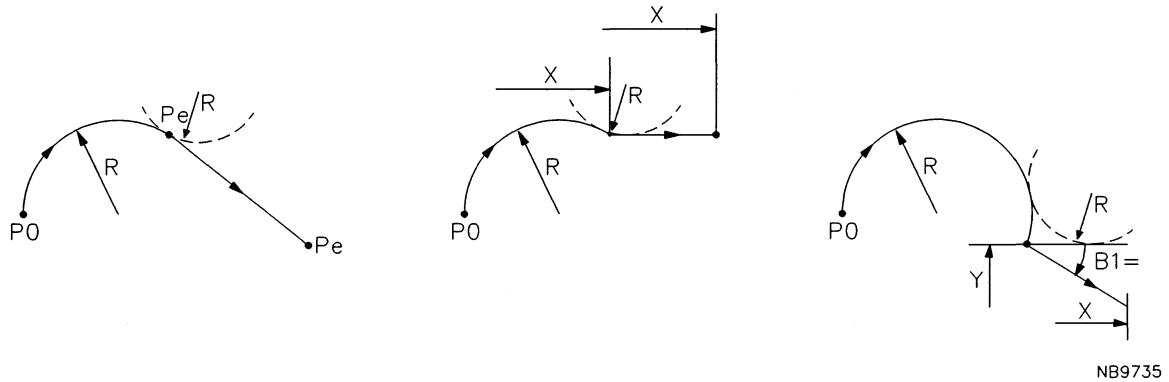
NB9698

N1 G2/G3 I.. J.. R.. J1=1/2

Refer to the section with the known start point for the formats of block N3

programmed intersection point

start point from N1 is known



N1 G2/G3 R.. [endpoint]
 N2 G3/G2 R..
 N3 G1 etc.

Refer to the section for programming the intersection point for the formats of block N3.

Note: A rounding can only be inserted if both the circle and the line are programmed with the endpoint, as indicated in the first format.

start point from N1 is not known

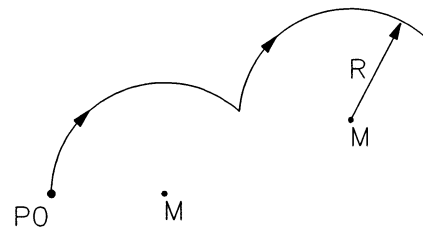
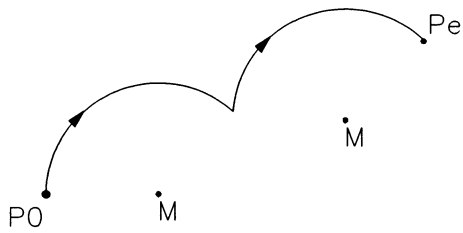
If the start point from N1 is not known, the centre point coordinates instead of the radius have to be programmed in block N1. So this block reads:

N1 G2/G3 I.. J.. [endpoint]
 N2 G3/G2 R..
 N3 etc.

Refer to the section with the known start point for the formats of block N3.

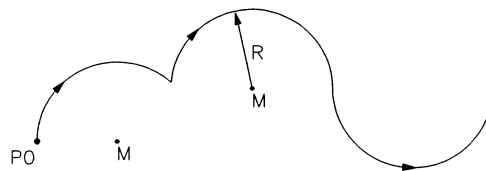
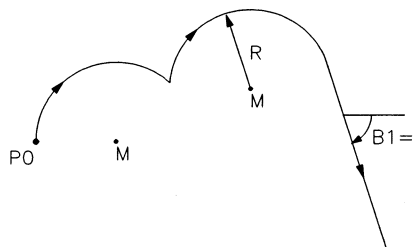
INTERSECTION POINT BETWEEN TWO CIRCLES To calculate the intersection point between two circles

start point from N1 is known



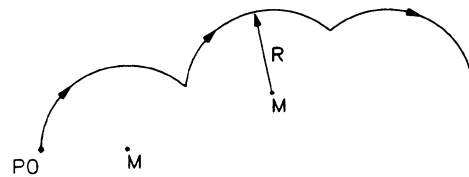
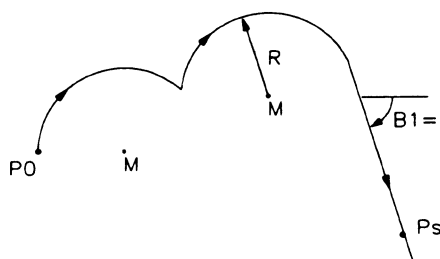
NB9699

N1 G2/G3 I.. J.. J1=1/2
N2 G2/G3 I.. J.. X.. Y..
or
N1 G2/G3 I.. J.. J1=1/2
N2 G2/G3 I.. J.. R..



NB9700

N1 G2/G3 I.. J.. J1=1/2
N2 G2/G3 I.. J.. R.. R1=0

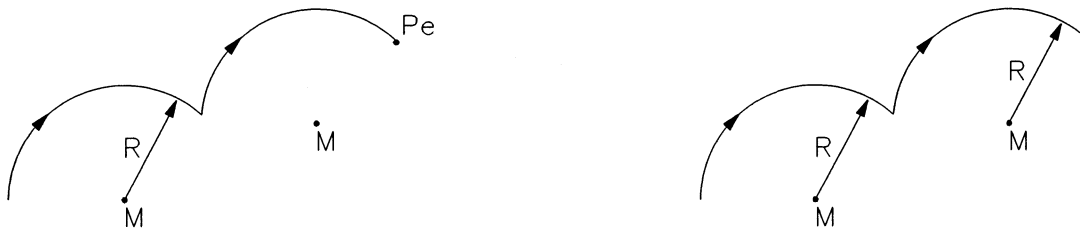


NB9701

N1 G2/G3 I.. J.. J1=1/2
N2 G2/G3 I.. J.. R.. J1=1/2

start point from N1 is not known

If the start point is not known, the radius value (R-word) should also be programmed in the block N1, thus this block reads:



NB9702

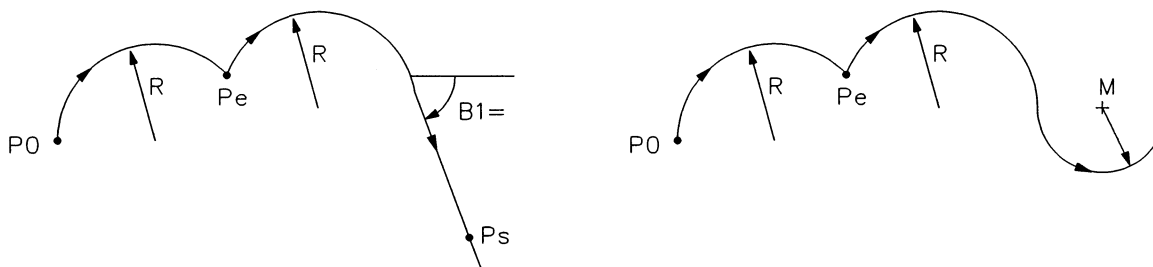
N1 G2/G3 I.. J.. R.. J1=1/2
The N2 block remains unchanged.

INTERSECTION POINT BETWEEN TWO CIRCLES PROGRAMMED AS END POINT

The intersection point between two circles is sometimes shown in the drawing and can be programmed directly. This point is assumed to be the start point of the next movement.

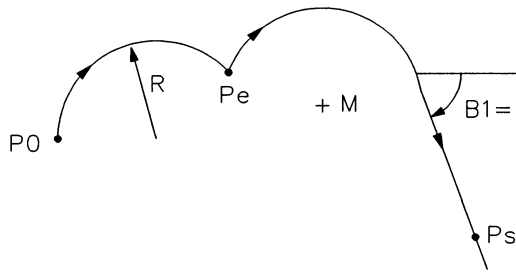
start point from N1 is not known

If the start point is not known, the radius value (R-word) should also be programmed in the block N1, thus this block reads:



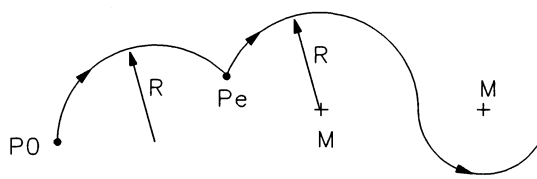
NB9736

N1 G2/G3 R.. [endpoint]
N2 G2/G3 R.. R1=0



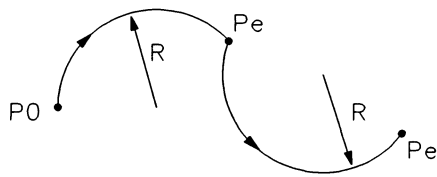
NB9737

N1 G2/G3 R.. [endpoint]
N2 G2/G3 I.. J.. {R1=0} {J1=1/2}



NB9738

N1 G2/G3 R.. [endpoint]
N2 G2/G3 I.. J.. R.. {R1=0} {J1=1/2}

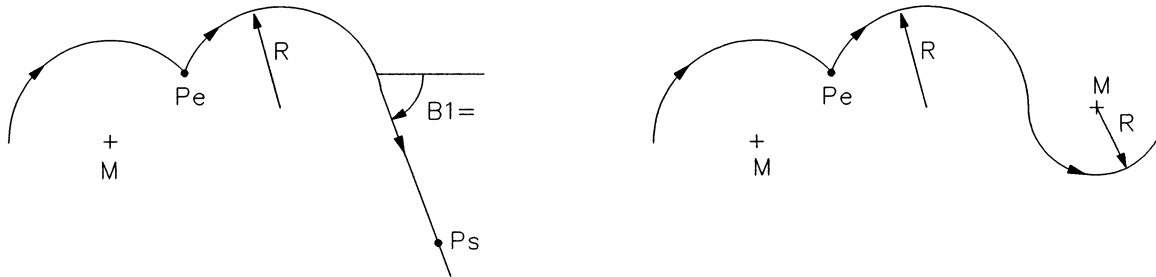


NB9739

N1 G2/G3 R.. [endpoint]
N2 G2/G3 R.. [endpoint]
or
N1 G2/G3 R.. [endpoint]
N2 G2/G3 I.. J.. [endpoint]

start point from N1 is not known

If the start point from N1 is not known, the centre point coordinates instead of the radius have to be programmed in block N1. So this block reads:



NB9740

N1 G2/G3 I.. J.. [endpoint]

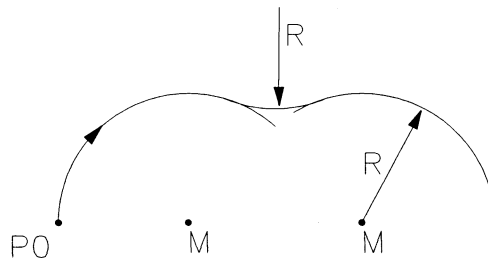
Block N2 from the mentioned cases remains the same.

ROUNDING BETWEEN TWO INTERSECTING CIRCLES

To insert a rounding between two intersecting circles:

calculated intersection point

start point from N1 is known



NB9703

N1 G2/G3 I.. J.. J1=1/2

N2 G3/G2 R..

N3 G2/G3 etc.

Refer to the section for calculating the intersection point for the formats of block N3.

start point from N1 is not known

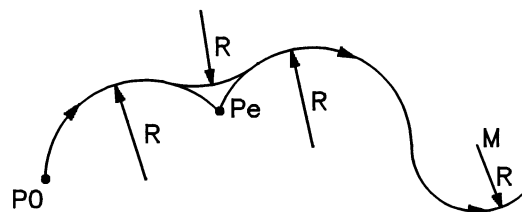
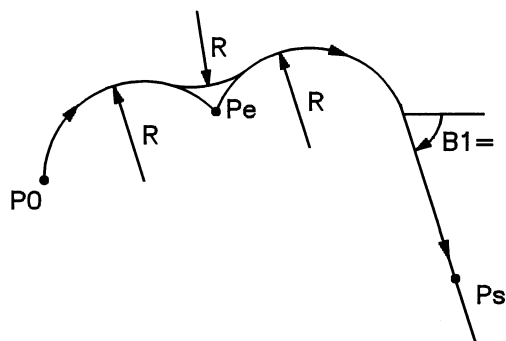
If the start point is not known, the radius value (R-word) should also be programmed in the block N1, thus this block reads:

N1 G2/G3 I.. J.. R.. J1=1/2

N2 G3/G2 R..

N3 G2/G3 etc.

Refer to the section with the known start point for the formats of block N3.

programmed intersection point**start point from N1 is known**

NB9700A

N1 G2/G3 R.. [endpoint]
 N2 G3/G2 R..
 N3 G2/G3 etc.

Refer to the section for programming the intersection point for the formats of block N3.

start point from N1 is not known

If the start point from N1 is not known, the centre point coordinates instead of the radius have to be programmed in block N1. So this block reads:

N1 G2/G3 I.. J.. [endpoint]
 N2 G3/G2 R..
 N3 G2/G3 etc.

Refer to the section with the known start point for the formats of block N3.

POINT OF TANGENCY**POINT OF TANGENCY INDICATOR (R1=)**

A special word R1=0 is used to indicate that a geometric element is tangent to the next one (connecting circles are disregarded), thus:

line tangent to circle
 circle tangent to line
 circle tangent to circle.

The word R1=0 is written in the block with the first element.

The point of tangency is chosen in such a way that the tool path is continue, that is to say the tool does not move backwards.

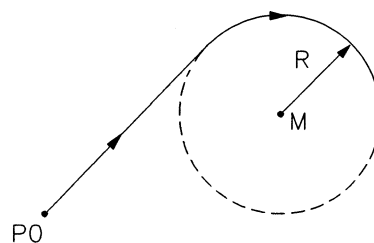
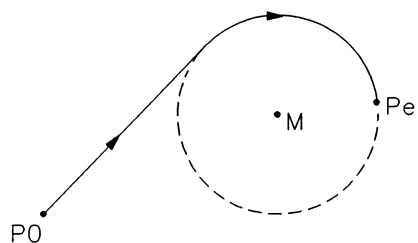
TANGENT LINE AND CIRCLE

To calculate the point of tangency between line and circle

If the start point of the line is known, two cases must be considered:

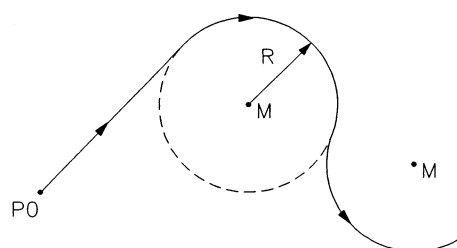
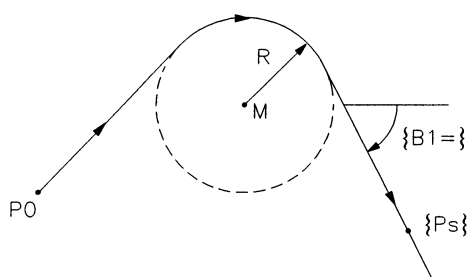
only the start point from N1 is known

The circle is defined by its centre point coordinates and the radius or end point. The following formats for line and circle can be used:



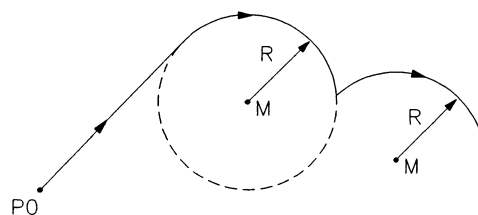
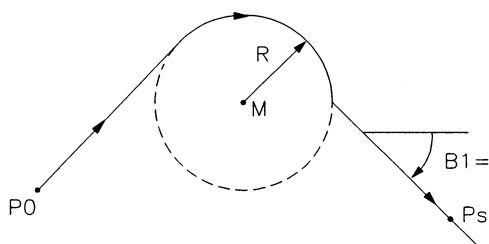
NB9704

N1 G1 R1=0
N2 G2/G3 I.. J.. X.. Y..
or
N1 G1 R1=0
N2 G2/G3 I.. J.. R..



NB9705

N1 G1 R1=0
N2 G2/G3 I.. J.. R.. R1=0

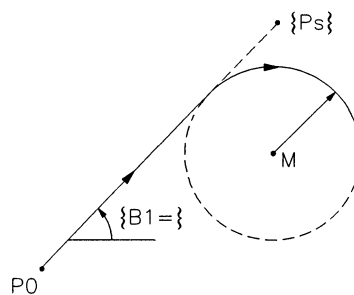
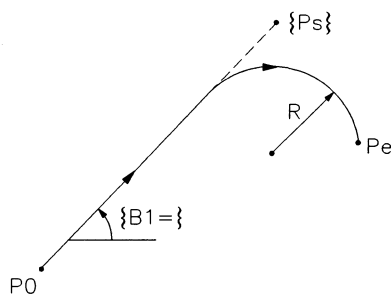


NB9706

N1 G1 R1=0
N2 G2/G3 I.. J.. R.. J1=1/2

start point from N1 and the angle with the axis or a support point on the line are known

From the circle in block N2 either the radius or the centre point coordinates must be calculated by the control. In this case the following formats are available:



NB9707

N1 G1 R1=0 B1=...
N2 G2/G3 R.. X.. Y..

or

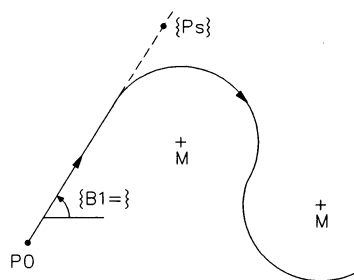
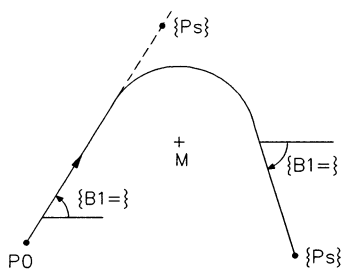
N1 G1 R1=0 B1=...
N2 G2/G3 I.. J..

or

N1 G1 R1=0 X.. Y.. I1=0
N2 G2/G3 R.. X.. Y..

or

N1 G1 R1=0 X.. Y.. I1=0
N2 G2/G3 I.. J..

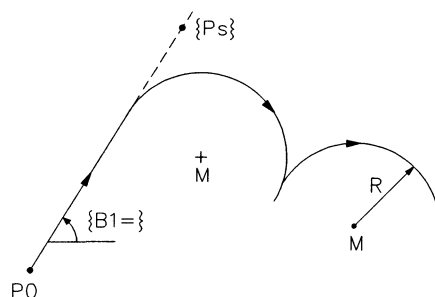
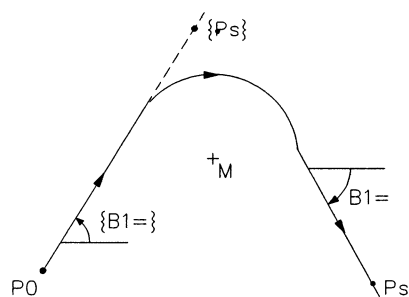


NB9708

N1 G1 R1=0 B1=...
N2 G2/G3 I.. J.. R1=0

or

N1 G1 R1=0 X.. Y.. I1=0
N2 G2/G3 I.. J.. R1=0

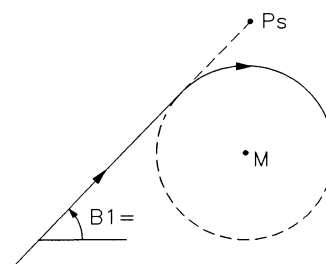
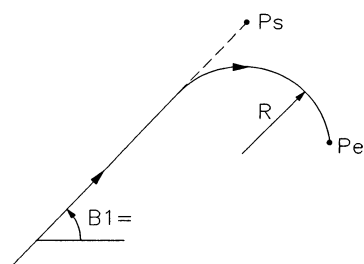


NB9709

N1 G1 R1=0 B1=...
 N2 G2/G3 I.. J.. J1=1/2
 or
 N1 G1 R1=0 X.. Y.. I1=0
 N2 G2/G3 I.. J.. J1=1/2

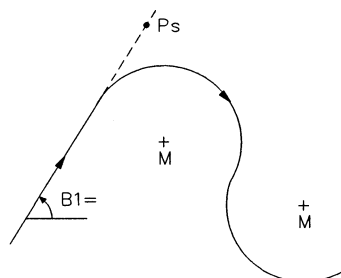
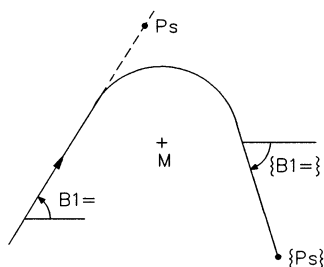
start point from N1 is not known

If the start point of the line is not known, the angle with the main axis and a support point on the line have to be programmed. For block N2 the formats from the second case are used, thus:



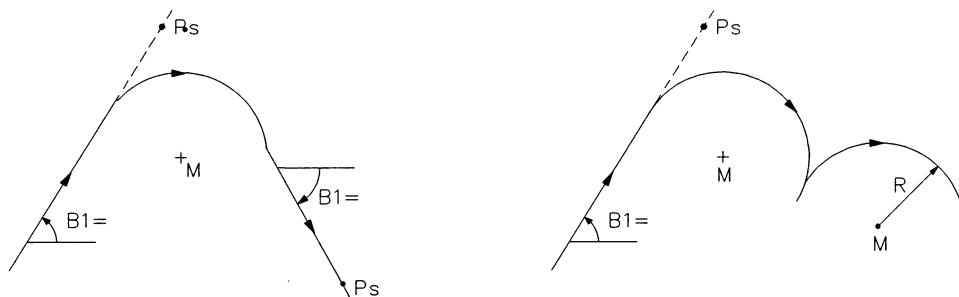
NB9710

N1 G1 R1=0 B1=.. X.. Y.. I1=0 or I1=±..
 N2 G2/G3 R.. X.. Y..
 or
 N1 G1 R1=0 B1=.. X.. Y.. I1=0 or I1=±..
 N2 G2/G3 I.. J..



NB9711

N1 G1 R1=0 B1=.. X.. Y.. I1=0 or I1=±..
 N2 G2/G3 I.. J.. R1=0

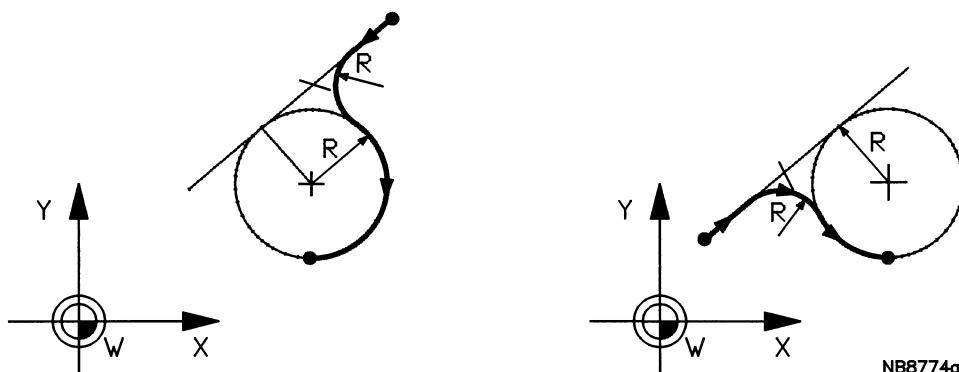


NB9712

N1 G1 R1=0 B1=.. X.. Y.. I1=0 or I1=±..
N2 G2/G3 I.. J.. J1=1/2

CONTINUOUS CONNECTING CIRCLE BETWEEN TANGENT LINE AND CIRCLE

To insert a connecting circle between a tangent line and a circle Only one connecting circle is possible. Its direction of rotation is opposite to the direction of rotation on the tangent circle.



NB8774a

start point from N1 is not known

N1 G1 R1=0 {B1=..}
N2 G2/G3 R..
N3 G3/G2 etc.
or
N1 G1 R1=0 {X.. Y.. I1=0}
N2 G2/G3 R..
N3 G3/G2 etc.

Refer to the previous sections with a unknown start point from N1 for the formats of block N3

Start point from N1 is not known

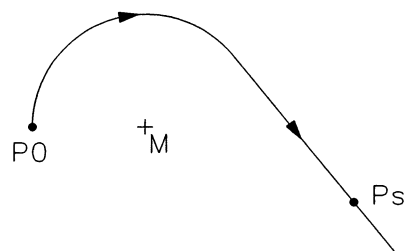
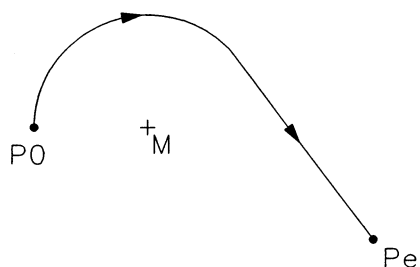
N1 G1 R1=0 B1=.. X.. Y.. I1=0 or I1=..
N2 G2/G3 R..
N3 G3/G2 etc.

Refer to the section with unknown start point for the formats of block N3

TANGENT CIRCLE AND LINE

To calculate the point of tangency between circle and line

start point from N1 and the end point or a support point of the line are known



NB9713

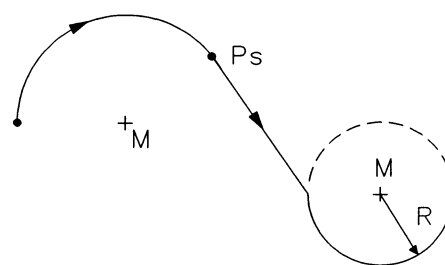
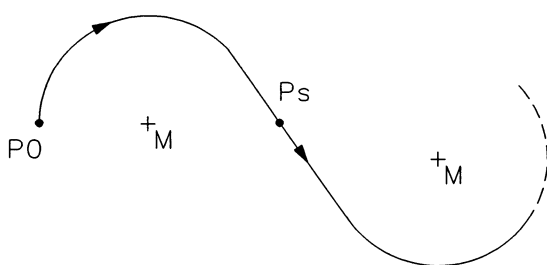
N1 G2/G3 I.. J.. R1=0

N2 G1 X.. Y..

or

N1 G2/G3 I.. J.. R1=0

N2 G1 X.. Y.. I1=0



NB9714

N1 G2/G3 I.. J.. R1=0

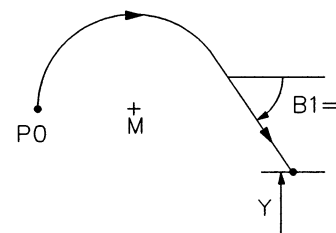
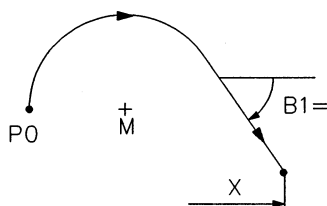
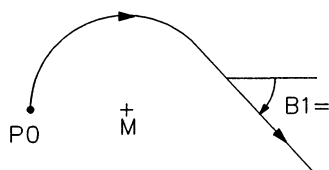
N2 G1 X.. Y.. I1=0 R1=0

or

N1 G2/G3 I.. J.. R1=0

N2 G1 X.. Y.. I1=0 J1=1/2

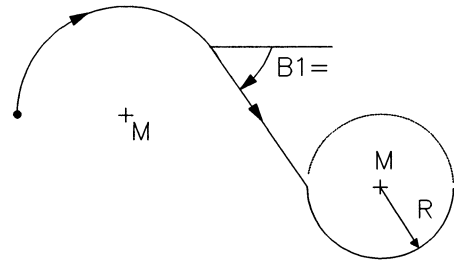
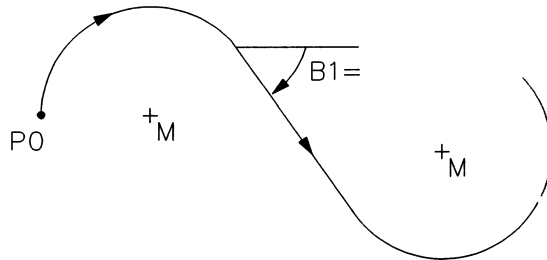
start point from N1 and the angle the line makes with the axis are known



NB9715

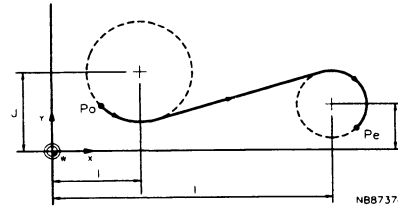
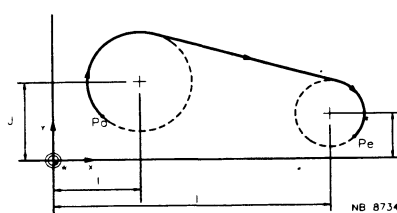
N1 G2/G3 I.. J.. R1=0

N2 G1 B1=..
or
N1 G2/G3 I.. J.. R1=0
N2 G1 B1=.. X..
or
N1 G2/G3 I.. J.. R1=0
N2 G1 B1=.. Y..



NB9716

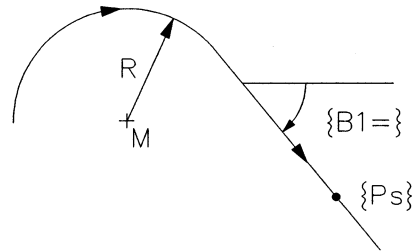
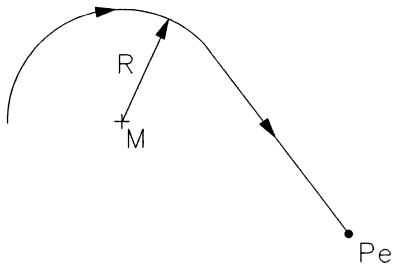
N1 G2/G3 I.. J.. R1=0
N2 G1 B1=.. R1=0
or
N1 G2/G3 I.. J.. R1=0
N2 G1 B1=.. J1=1/2
common tangent line of two circles



N1 G2/G3 I.. J.. R1=0
N2 G1 R1=0
N3 G2/G3 I.. J.. R..
or
N1 G2/G3 I.. J.. R1=0
N2 G1 R1=0
N3 G2/G3 I.. J.. X.. Y..

start point from N1 is not known

If the start point is not known, the radius value (R-word) should also be programmed in the block N1, thus this block reads:



NB9717

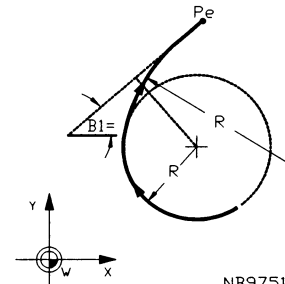
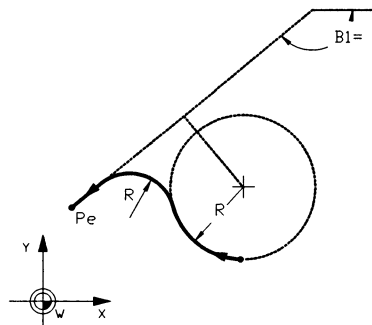
N1 G2/G3 I.. J.. R.. R1=0

The other blocks from the mentioned cases remain the same

CONTINUOUS CONNECTING CIRCLE BETWEEN TANGENT CIRCLE AND LINE

To insert a connecting circle between a tangent circle and line.

Only one connecting circle is possible. Its direction of rotation is opposite to the direction of rotation on the tangent circle



NB9751

start point from N1 is known

N1 G2/G3 I.. J.. R1=0

N2 G3/G2 R..

N3 G1 etc.

Refer to the previous sections with a known start point for the formats of block N3.

start point from N1 is not known

If the start point is not known, the radius value (R-word) should also be programmed in the block N1, thus this block reads:

N1 G2/G3 I.. J.. R.. R1=0

N2 G3/G2 R..

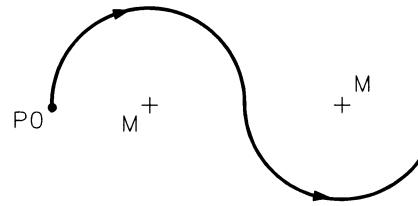
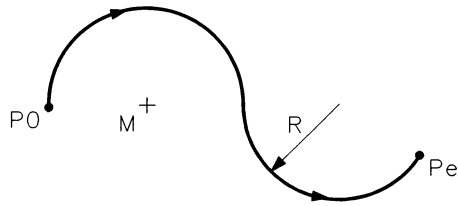
N3 G1 etc.

Refer to the section with unknown start point for the formats of block N3.

TANGENT CIRCLE AND LINE

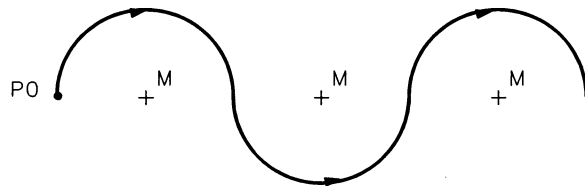
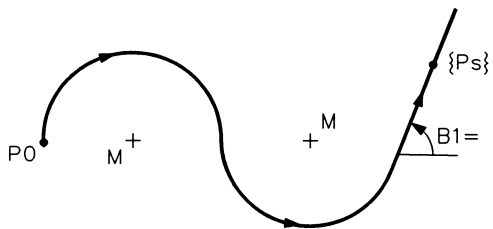
To calculate the point of tangency between circle and line

start point from N1 is known



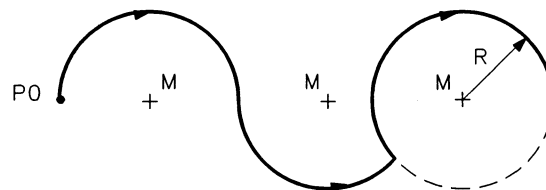
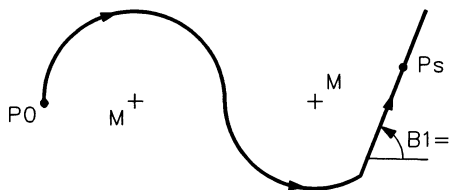
NB9718

N1 G2/G3 I.. J.. R1=0
N2 G2/G3 R.. X.. Y..
or
N1 G2/G3 I.. J.. R1=0
N2 G2/G3 I.. J..



NB9719

N1 G2/G3 I.. J.. R1=0
N2 G2/G3 I.. J.. R1=0

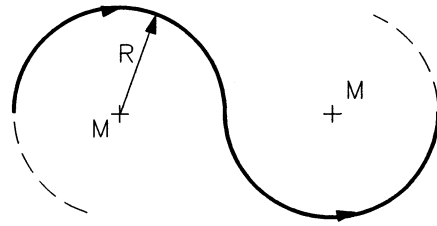
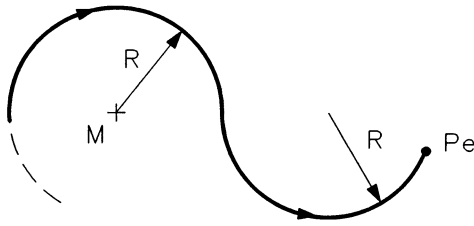


NB9720

N1 G2/G3 I.. J.. R1=0
N2 G2/G3 I.. J.. J1=1/2

start point from N1 is not known

If the start point is not known, the radius value (R-word) should also be programmed in the block N1, thus this block reads:



NB9721

N1 G2/G3 I.. J.. R.. R1=0

The other blocks from the mentioned cases remain the same.

CONTINUOUS CONNECTING CIRCLE BETWEEN TWO TANGENT CIRCLES

To insert a connecting circle between two tangent circles.

The connecting circle can be outside both circles or surround them. The programming of the direction of rotation on the three circles is:

1. Connecting circle tangent on the outside of both circles

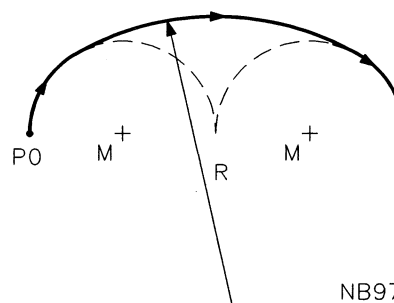
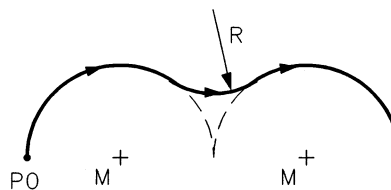
G2, G3, G2 or G3, G2, G3

2. Connecting circle surrounds both circles

G2, G2, G2 or G3, G3, G3

For both cases the following formats are available. The right combination of the direction of rotation on the three circles has to be chosen by the programmer.

start point from N1 is known



NB9722

N1 G2/G3 I.. J.. R1=0

N2 G3/G2 R..

N3 G2/G3 etc.

Refer to the previous sections with a known start point for the formats of block N3.

start point from N1 is not known

N1 G2/G3 I.. J.. R.. R1=0

The other blocks from the mentioned cases remain the same.

CONTINUOUS CONNECTING CIRCLE BETWEEN ELEMENTS WHICH DO NOT MEET**LINE AND CIRCLE**

To insert a connecting circle between a line which does not meet a circle. Two connecting circles are possible:

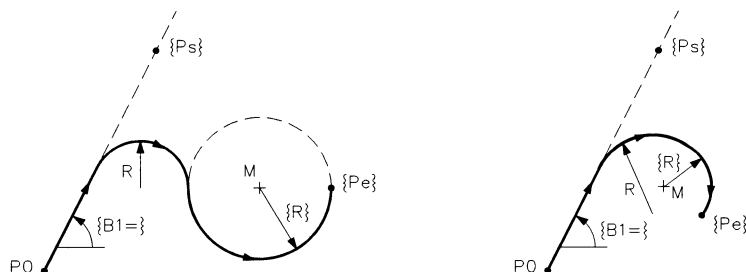
1. Connecting circle is tangent on the outside of the circle

G1, G2, G3 or G1, G3, G2

2. Connecting circle surrounds the circle

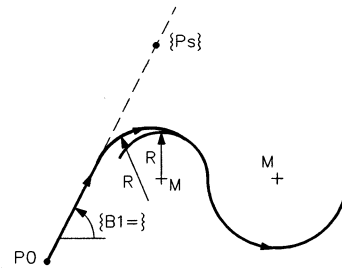
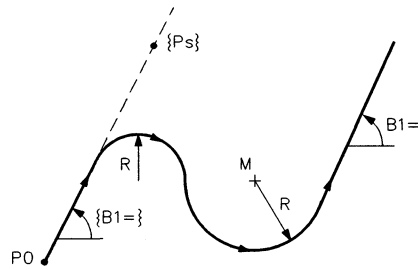
G1, G2, G2 or G1, G3, G3

For both cases the following formats are available. The right combination of the direction of rotation on the circles has to be chosen by the programmer.

start point from N1 is known

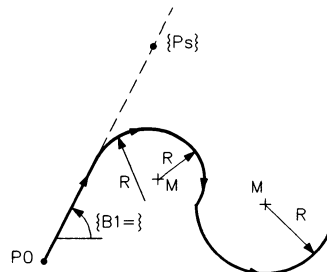
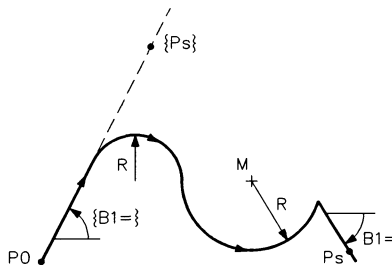
NB9723

N1 G1 B1=..
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. X.. Y..
 or
 N1 G1 B1=..
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. R..
 or
 N1 G1 X.. Y.. I1=0
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. X.. Y..
 or
 N1 G1 X.. Y.. I1=0
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. R..



NB9741

N1 G1 B1=..
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. R.. R1=0
 N1 G1 X.. Y.. I1=0
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. R.. R1=0



NB9742

N1 G1 B1=..
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. R.. J1=1/2
 or
 N1 G1 X.. Y.. I1=0
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. R.. J1=1/2

start point from N1 is not known

N1 G1 B1=... X.. Y.. I1=0 or I1=..

The other blocks from the mentioned cases remain the same.

CIRCLE AND LINE

To insert a connecting circle between a circle and a line which do not meet each other. Two connecting circles are possible:

1. Connecting circle is tangent on the outside of the circle

G2, G3, G1 or G3, G2, G1

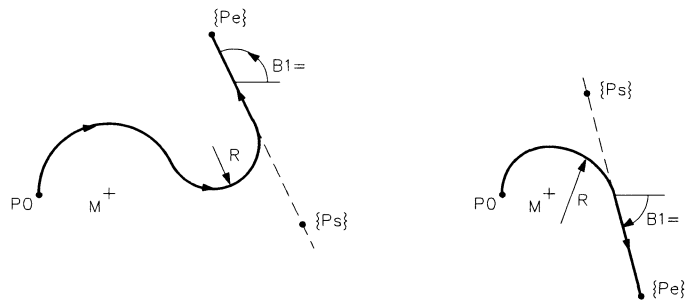
2. Connecting circle surrounds the circle

G2, G2, G1 or G3, G3, G1

For both cases the following formats are available. The right combination of the direction of rotation

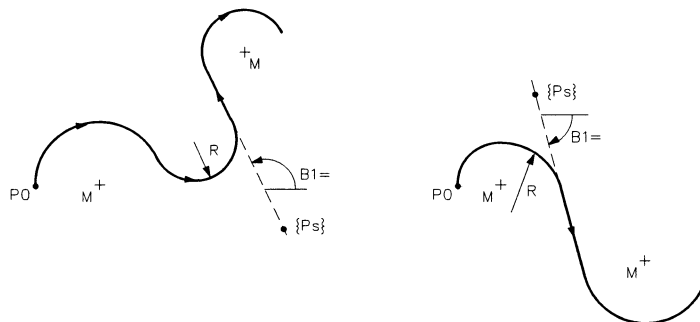
on the circles has to be chosen by the programmer.

start point from N1 is known



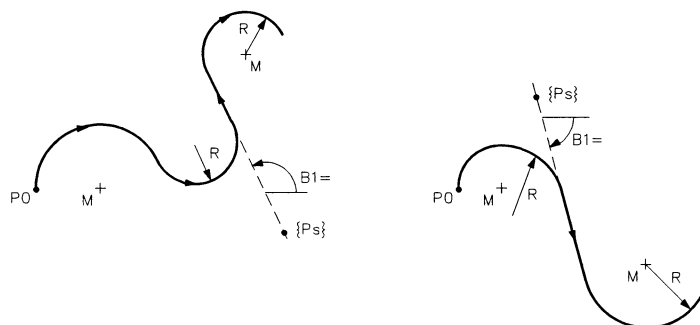
NB9743

N1 G2/G3 I.. J..
 N2 G3/G2 R..
 N3 G1 B1=.. X.. Y..
 or
 N1 G2/G3 I.. J..
 N2 G3/G2 R..
 N3 G1 B1=.. X.. Y.. I1=0



NB9744

N1 G2/G3 I.. J..
 N2 G3/G2 R..
 N3 G1 B1=.. X.. Y.. I1=0 R1=0



NB9745

N1 G2/G3 I.. J..
 N2 G3/G2 R..
 N3 G1 B1=.. X.. Y.. I1=0 J1=1/2

start point from N1 is not known

N1 G2/G3 I.. J.. R..

The other blocks from the mentioned cases remain the same.

TWO CIRCLES OUTSIDE EACH OTHER

To insert a connecting circle between two circles outside each other which do not meet. The direction of rotation on the three circles indicates the type of connecting circle, thus:

1. Connecting circle tangent on the outside of both circles

G2, G3, G2 or G3, G2, G3

2. Connecting circle surrounds both circles

G2, G2, G2 or G3, G3, G3

3. Connecting circle is tangent on the outside of the first circle and surrounds the second circle

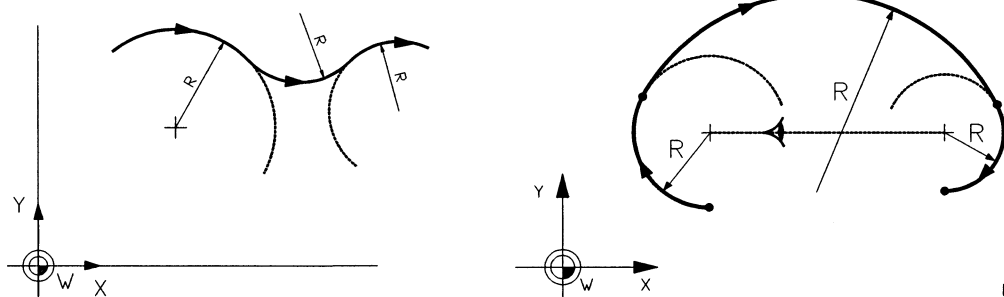
G2, G3, G3 or G3, G2, G2

4. Connecting circle surrounds the first circle and is tangent on the outside of the second circle

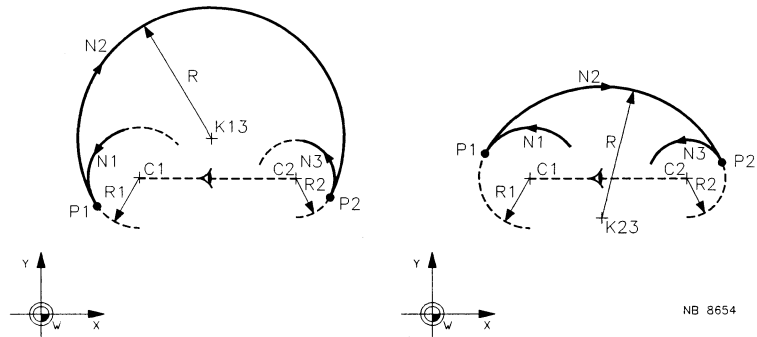
G2, G2, G3 or G3, G3, G2

For all four cases the following formats are available. Any combination of the direction of rotation on the three circles is possible:

start point from N1 is known



NB9752



NB 8654

N1 G2/G3 I.. J..
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. X.. Y..
 or
 N1 G2/G3 I.. J..
 N2 G3/G2 R..
 N3 G2/G3 I.. J.. R.. {R1=0} {J1=1/2}

start point from N1 is not known

N1 G2/G3 I.. J.. R..

The other blocks from the mentioned cases remain the same.

ONE CIRCLE INSIDE THE OTHER ONE

To insert a connecting circle between a circle inside the other one which do not meet. The direction of rotation on the three circles indicates the type of connecting circle, thus:

1. Connecting circle tangent on the outside of inner circle

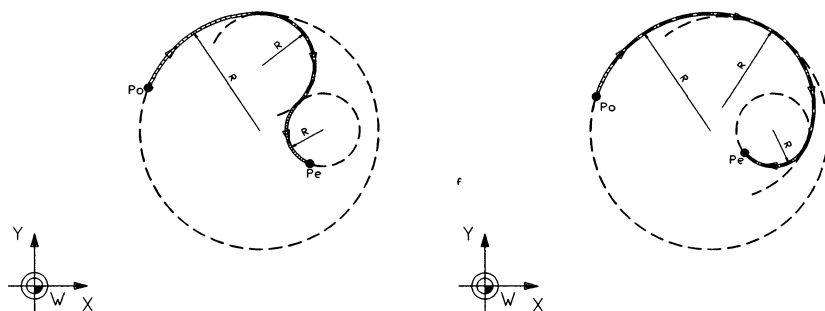
G2, G2, G3 or G3, G3, G2

2. Connecting circle surrounds the inner circle

G2, G2, G2 or G3, G3, G3

For both cases the following formats are available:

start point from N1 is known



NB9754

N1 G2/G3 I.. J..
 N2 G2/G3 R..
 N3 G2/G3 I.. J.. X.. Y..
 or
 N1 G2/G3 I.. J..

N2 G2/G3 R..
 N3 G2/G3 I.. J.. R.. {R1=0} {J1=1/2}

start point from N1 is not known

N1 G2/G3 I.. J.. R..

The other blocks from the mentioned cases remain the same.

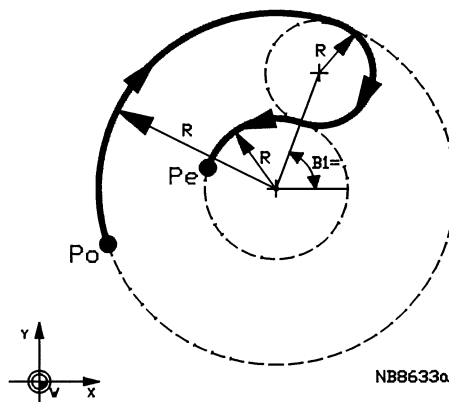
CONCENTRIC CIRCLES

Two concentric circles are a very special case of one circle inside the other one. In this case the centre points of both circles coincide, so some additional information has to be programmed to let the control calculate the connecting circle.

The word B1=.. which indicate the angle with the main axis of the line through the centre point of the concentric circles and the connecting circle, is used as additional information and has to be inserted in the block with the connecting circle.

For both cases the following formats are available:

start point from N1 and radius of the connecting circle are known



N1 G2/G3 I.. J..
 N2 G2/G3 R.. B1=..
 N3 G2/G3 I.. J.. {R1=0} {J1=1/2}

start point from N1 and radius of the second circle are known

In this case the radius of the connecting circle is calculated by the control.

N1 G2/G3 I.. J..
 N2 G2/G3 B1=..
 N3 G2/G3 I.. J.. X.. Y..
 or
 N1 G2/G3 I.. J..
 N2 G2/G3 B1=..
 N3 G2/G3 I.. J.. R.. {R1=0} {J1=1/2}

start point from N1 is not known

N1 G2/G3 I.. J.. R..

The other blocks from the mentioned cases remain the same.

106.1 Geometric calculations with non-continuous movements G64

CONTENTS OF THIS FORMAT SECTION

ROUNDING OR CONNECTING CIRCLE INDICATOR (K1=)

ROUNDING WITH INTERSECTION POINTS

ROUNDING BETWEEN INTERSECTING STRAIGHT LINES

ROUNDING BETWEEN INTERSECTING LINE AND CIRCLE

ROUNDING BETWEEN INTERSECTING CIRCLE AND LINE

ROUNDING BETWEEN TWO INTERSECTING CIRCLES

TANGENT LINES (R1=)

CONNECTING CIRCLE BETWEEN TANGENT LINE AND CIRCLE OR V.V.

CONNECTING CIRCLE BETWEEN A LINE WHICH DO NOT MEET A CIRCLE

CONNECTING CIRCLE BETWEEN CIRCLES OUTSIDE EACH OTHER

CONNECTING CIRCLE BETWEEN TWO CIRCLES ONE INSIDE THE OTHER

CONNECTING CIRCLE WITH TWO CONCENTRIC CIRCLES

ROUNDING OR CONNECTING CIRCLE INDICATOR (K1=)

A special indicator (K1 =) is introduced to program which rounding or connecting circle is to be used.

For a rounding the value of K1= can be 1, 2, 3, or 4. Refer to the section ROUNDING WITH INTERSECTION POINTS for the meaning of these values. If with a rounding a wrong value is programmed, an error message is displayed.

For a connecting circle the indicator K1 = has two digits:

the first digit can have the value 1 or 2

=1 the left connecting circle

=2 the right connecting circle

the second digit indicates which connecting circle is meant and can have the values

=0 with a line tangent to a circle or v.v.

=0 or 1 with a line which does not meet a circle or v.v.

=2 to 7 with circular elements

Refer to the proper sections for programming a connecting circle to see the meaning of left and right and of the second digit.

ROUNDING WITH INTERSECTION POINTS

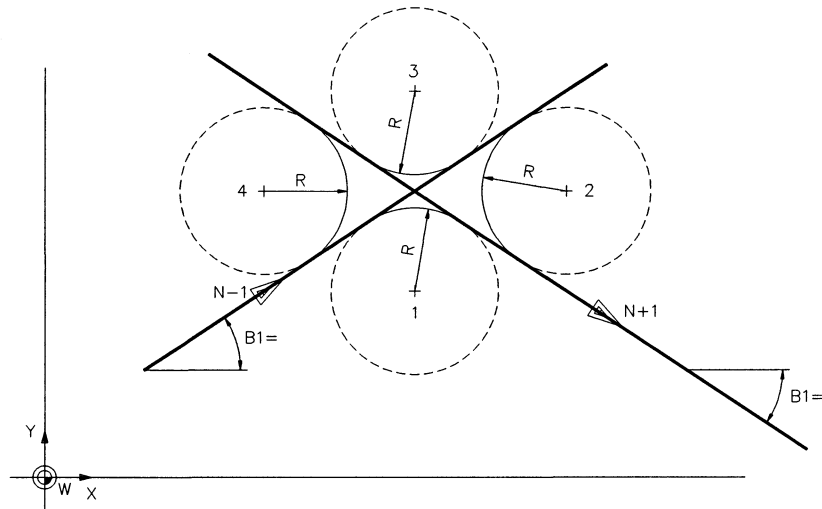
Between two intersecting elements a rounding can be inserted. In general four circles are possible which are numbered 1 to 4 and programmed with the word K1 = . The centre points of the circles with the numbers 1 and 2 are lying at the right from the first geometry element, when looking in the direction of the tool movement.

With the words K1 =2 or K1 =3 the contour intersects itself.

Note: If the word K1= is not programmed, a default value is used. This is K1=1 or K1=4 depending on the direction of movement on the second element.

ROUNDING BETWEEN INTERSECTING STRAIGHT LINES

To insert a rounding between two straight lines. The rounding can have any direction of rotation, thus G2 or G3.



NB6946

start point from N1 is known

N1 G1 B1=..
 N2 G2/G3 R.. K1=1/2/3/4
 N3 G1 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

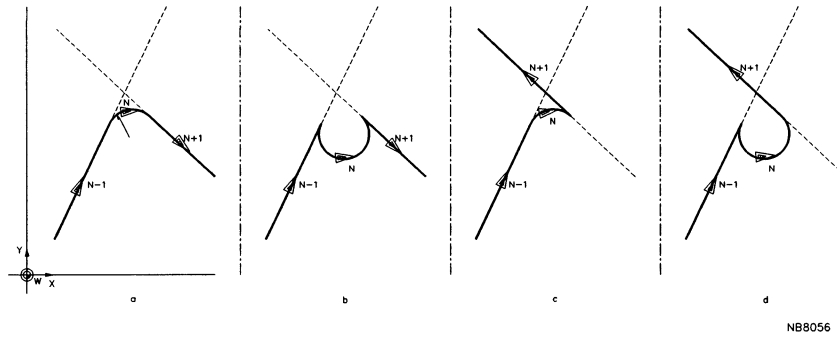
start point from N1 is not known

If the start point from N1 is not known, both the angle and a support point have to be programmed in block N1. So this block reads:

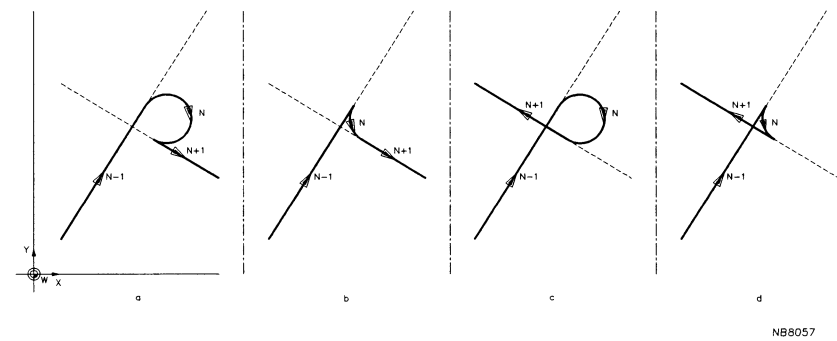
N1 G1 B1=.. X.. Y.. I1=0 or I1=..
 N2 G2/G3 R.. K1=1/2/3/4
 N3 G1 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

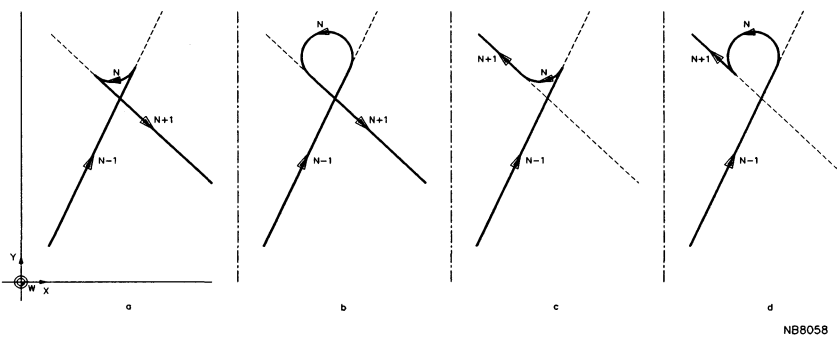
In the following illustrations the possibilities are shown with a circular connection between two straight lines.



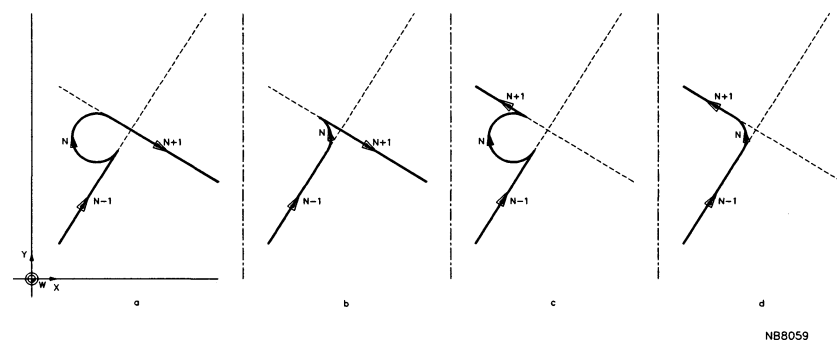
A circular connection with $K1=1$



A circular connection with $K1=2$



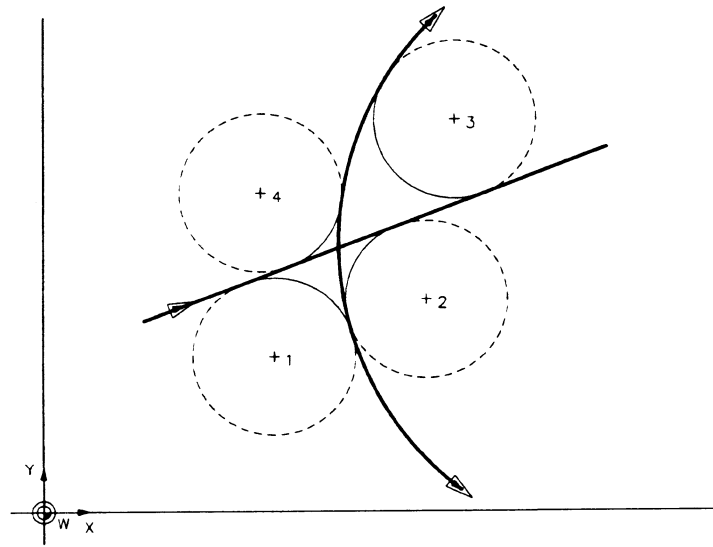
A circular connection with $K1=3$



A circular connection with $K1=4$

ROUNDING BETWEEN INTERSECTING LINE AND CIRCLE

To insert a rounding between an intersecting line and a circle. The rounding can have any direction of rotation, thus G2 or G3.



NB6909

start point from N1 is known

N1 G1 B1=.. J1=1/2
 N2 G3/G2 R.. K1=1/2/3/4
 N3 G2/G3 etc.

start point from N1 is not known

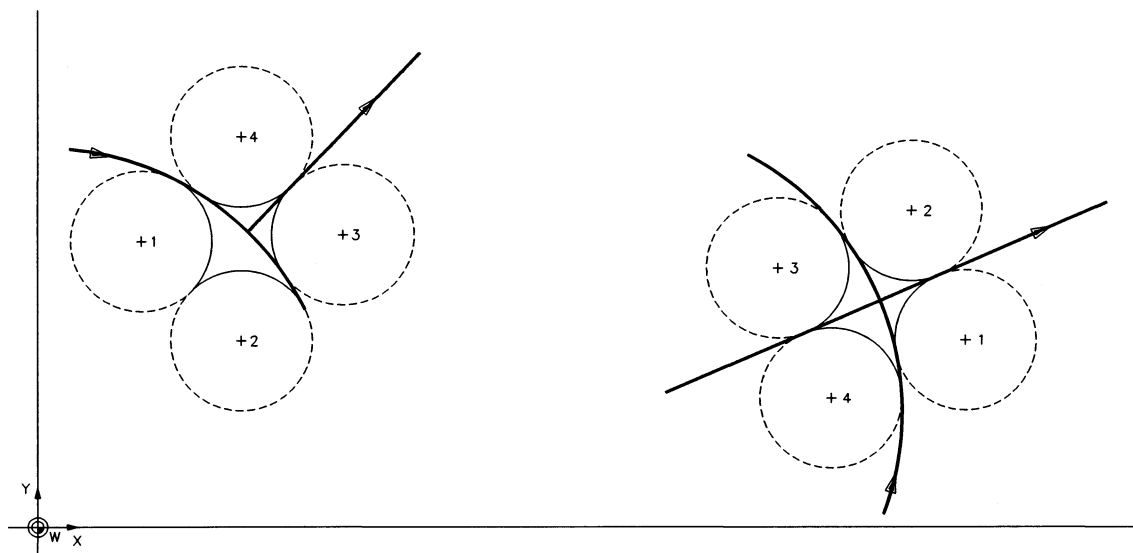
If the start point from N1 is not known, both the angle and a support point have to be programmed in block N1. So this block reads:

N1 G1 B1=.. X.. Y.. I1=0 or I1=..± J1=1/2
 N2 G3/G2 R.. K1=1/2/3/4
 N3 G2/G3 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

ROUNDING BETWEEN INTERSECTING CIRCLE AND LINE

To insert a rounding between an intersecting circle and a line. The rounding can have any direction of rotation, thus G2 or G3.



NB6910

start point from N1 is known

N1 G2/G3 I.. J.. J1=1/2
 N2 G3/G2 R.. K1=1/2/3/4
 N3 G1 etc.

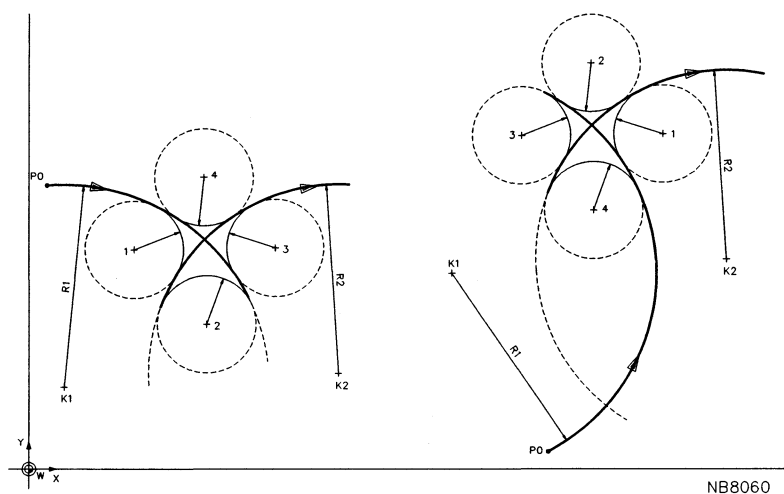
start point from N1 is not known

N1 G2/G3 I.. J.. R.. J1=1/2
 N2 G3/G2 R.. K1=1/2/3/4
 N3 G1 etc.

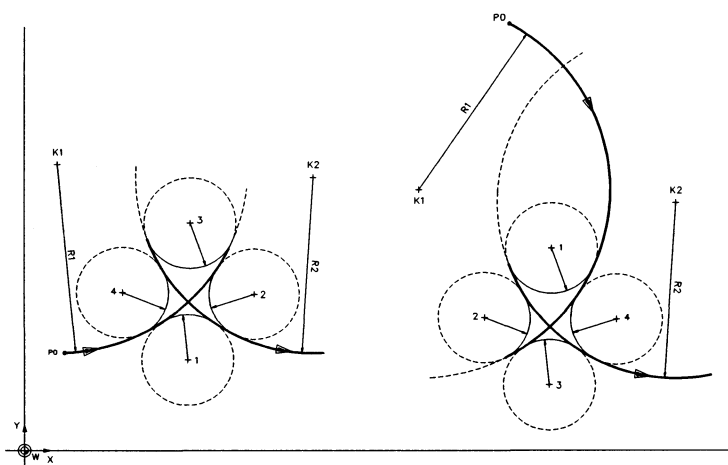
Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

ROUNDING BETWEEN TWO INTERSECTING CIRCLES

To insert a rounding between two intersecting circles. The rounding can have any direction of rotation, thus G2 or G3.



NB8060



NB8061

start point from N1 is known

N1 G2/G3 I.. J.. J1=1/2
 N2 G3/G2 R.. K1=1/2/3/4
 N3 G2/G3 etc.

start point from N1 is not known

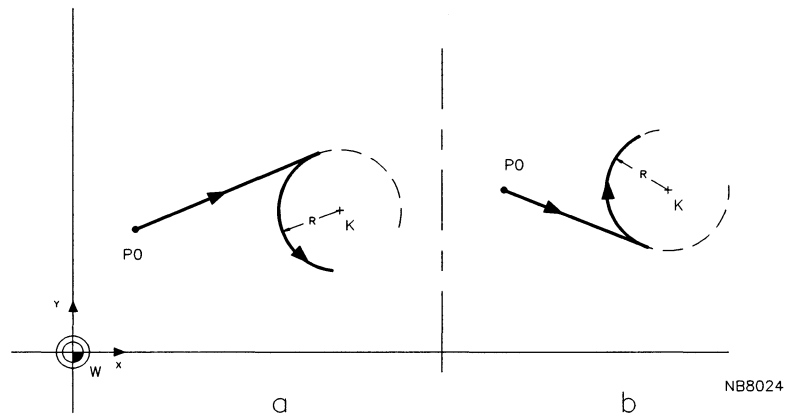
N1 G2/G3 I.. J.. R.. J1=1/2
 N2 G3/G2 R.. K1=1/2/3/4
 N3 G2/G3 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

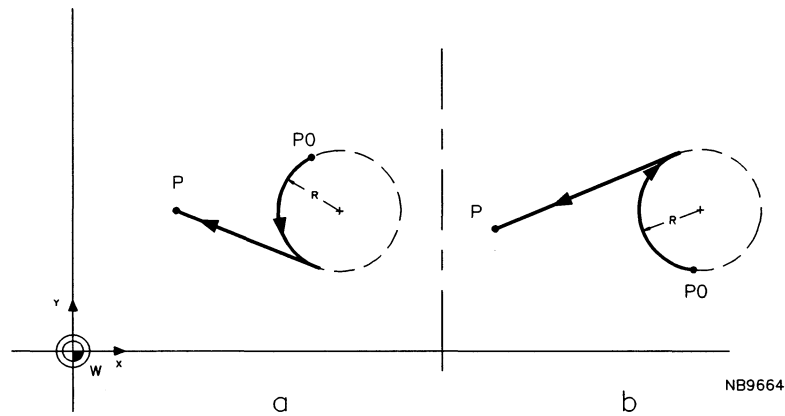
TANGENT LINES (R1=)

From a point two lines can be drawn tangent to a circle. With the word R1= in the block with the tangent element is indicated which tangent line should be used:

- a. R1=1: the left tangent line
- b. R1=2: the right tangent line



NB8024



NB9664

Left and right are determined with a movement from:

- line to circle** by looking from start point to centre point
- circle to line** by looking from centre point to end point

The word R1=1 or R1=2 is programmed in the same way as explained for R1=0 in the continuous section.

Note: With R1=0 the control determines automatically which tangent line keeps the movement continuous, so the tool does not move backwards.

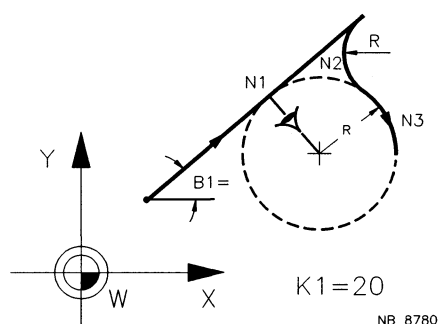
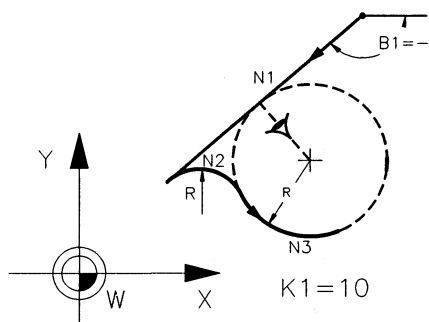
CONNECTING CIRCLE BETWEEN A LINE TANGENT TO A CIRCLE OR V.V.

If a line is tangent to a circle two connecting circles are possible, one left of the line through the centre of the circle and perpendicular to the line and one to the right of that line.

The left circle is programmed with the indicator K1=10 and the right circle with K1=20.

The formats are:

Line tangent to circle



NB 8780

start point from N1 is known

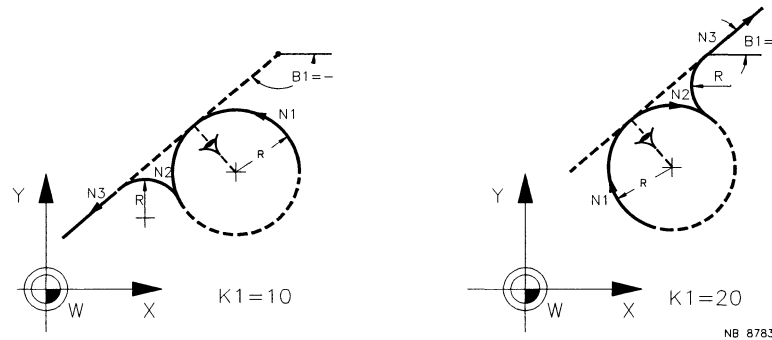
N1 G1 R1=1/2
N2 G2/G3 R.. K1=10/20
N3 G2/G3 etc.

start point from N1 is not known

N1 G1 B1=.. R1=1/2
N2 G2/G3 R.. K1=10/20
N3 G2/G3 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

Circle tangent to line



start point from N1 is known

N1 G2/G3 I.. J.. R1=1/2
 N2 G2/G3 R.. K1=10/20
 N3 G1 etc.

start point from N1 is not known

N1 G2/G3 I.. J.. R.. R1=1/2
 N2 G2/G3 R.. K1=10/20
 N3 G1 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

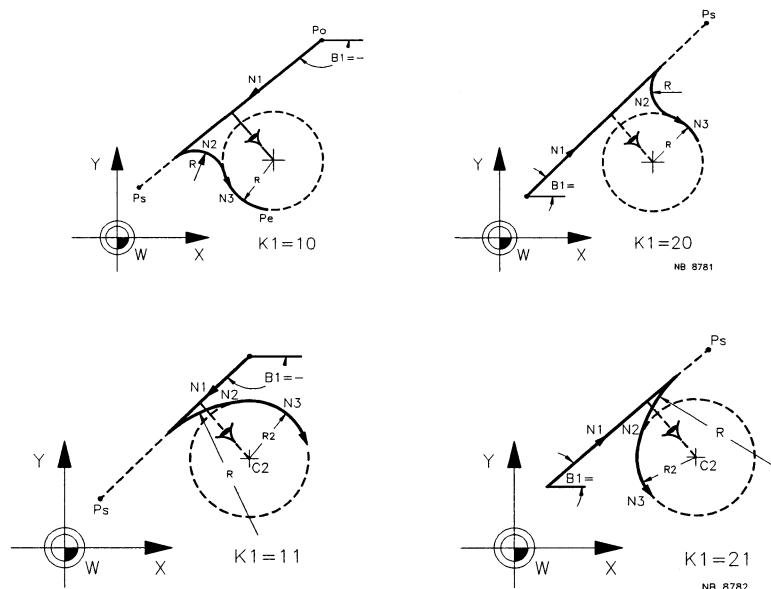
CONNECTING CIRCLE BETWEEN A LINE WHICH DOES NOT MEET A CIRCLE

If a line does not meet a circle, two connecting circles are possible on the left of the line through the centre and perpendicular to the line. The same two circles are also possible to the right of that line. One circle touches the circle on the outside. The left circle is programmed with the word K1 =10 and the same circle on the right with K1 =20.

The second connecting circle surrounds the circle. In this case the left circle is programmed with the word K1=11 and the same circle on the right with K1 =21.

The formats are:

Line and circle



start point from N1 is known

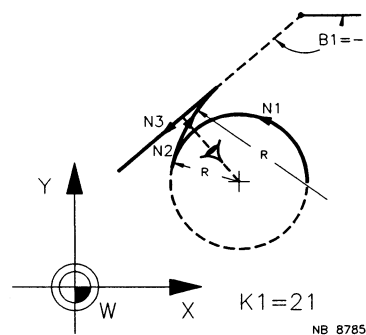
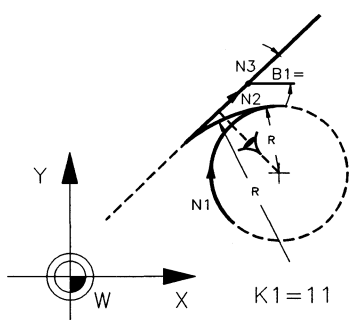
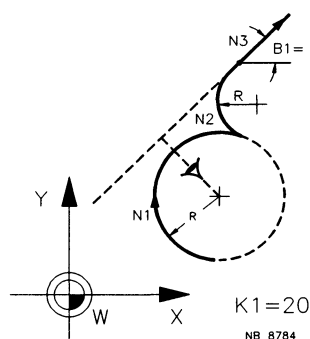
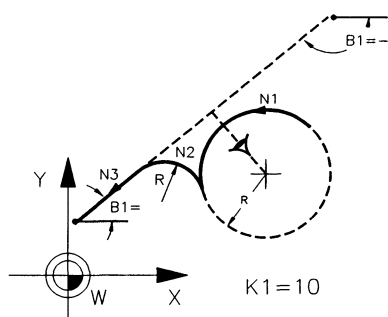
N1 G1 B1=.. {X.. Y.. I1=0}
 N2 G2/G3 R.. K1=10/11 or K1=20/21
 N3 G2/G3 etc.

start point from N1 is not known

N1 G1 B1=.. X.. Y.. I1=0
 N2 G2/G3 R.. K1=10/11 or K1=20/21
 N3 G2/G3 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

Circle and line



start point from N1 is known

N1 G2/G3 I.. J.. R1=1/2
N2 G2/G3 R.. K1=10/11 or K1=20/21
N3 G1 etc.

start point from N1 is not known

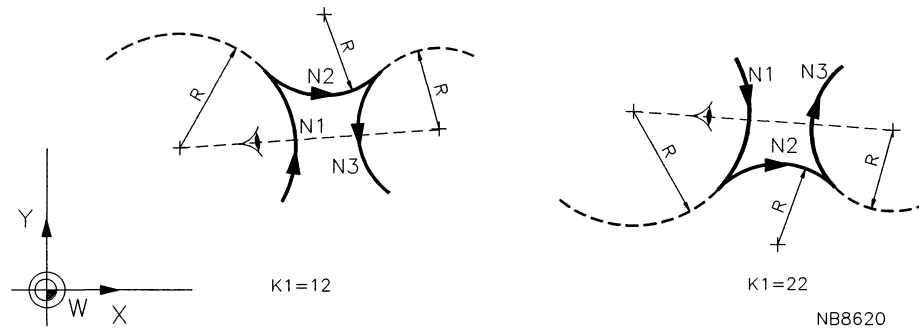
N1 G2/G3 I.. J.. R.. R1=1/2
N2 G2/G3 R.. K1=10/11 or K1=20/21
N3 G1 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

CONNECTING CIRCLE BETWEEN CIRCLES OUTSIDE EACH OTHER

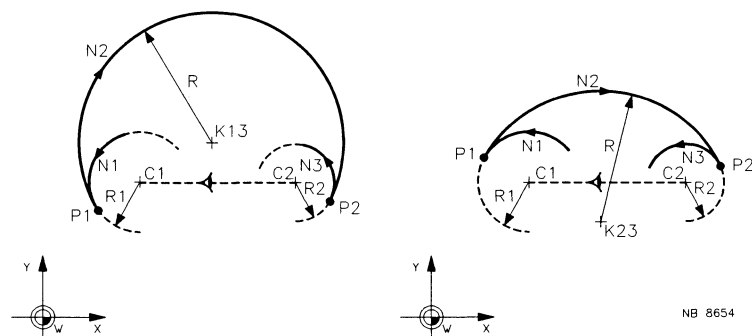
Four types of connecting circles on the left of the line through the centre points are possible with two circles which do not meet and are outside each other. The first two types are also possible with tangent circles. The same four types can be found on the right of the line through the centres.

The word K1=.. for a connecting circle outside both circles is:



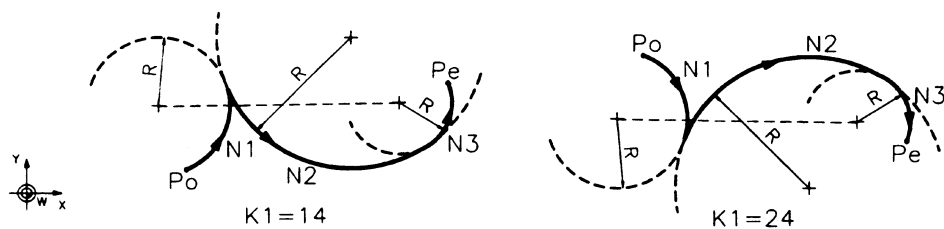
- for the **left** connecting circle: K1 =12
- for the **right** connecting circle: K1 =22

The word K1=.. for a connecting circle **surrounding both circles** is:



- for the **left** connecting circle: K1=13
- for the **right** connecting circle: K1=23

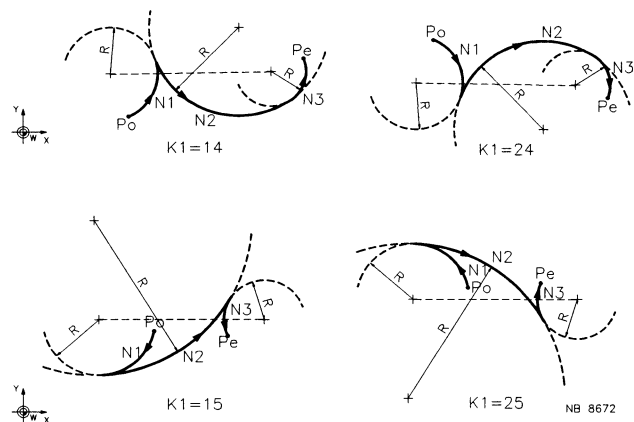
The word K1 =.. for a connecting circle **outside the first circle** is:



(NB8672)

- for the **left** connecting circle: K1 =14
- for the **right** connecting circle: K1=24

The word K1 =.. for a connecting circle **surrounding the first circle** is:



- for the **left** connecting circle: K1=15
- for the **right** connecting circle: K1 =25

Note: It depends on the programmed direction of movement (G2 and G3) on the three circles which default value for K1 = is used by the control.

The formats are:

start point from N1 is known

N1 G2/G3 I.. J.. R1=1/2
 N2 G2/G3 R.. K1=12/13/14/15 or K1=22/23/24/25
 N3 G2/G3 etc.

start point from N1 is not known

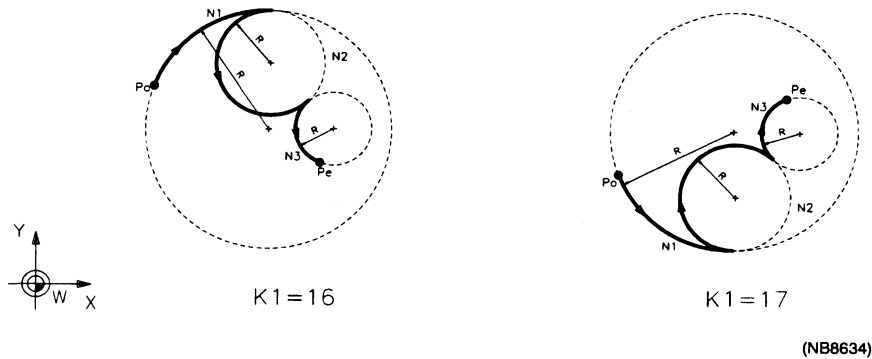
N1 G2/G3 I.. J.. R.. R1=1/2
 N2 G2/G3 R.. K1=12/13/14/15 or K1=22/23/24/25
 N3 G2/G3 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

CONNECTING CIRCLE BETWEEN TWO CIRCLES ONE INSIDE THE OTHER

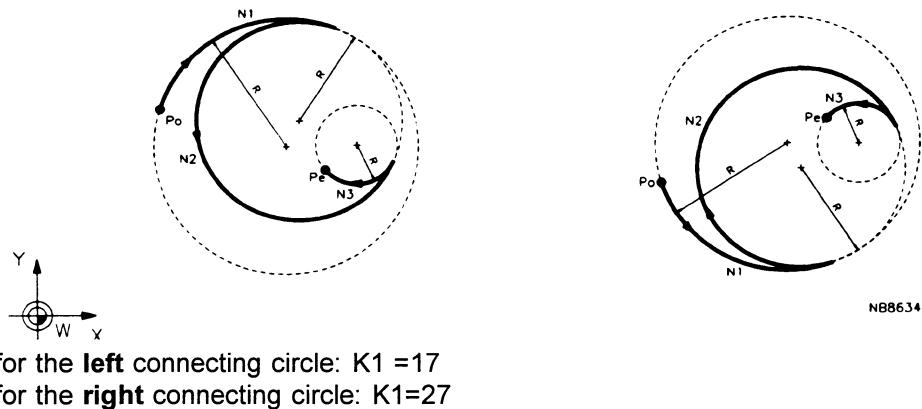
Two types of connecting circles on the left of the line through the centre points are possible with two circles of which one lies inside the other one. The same two types can be found on the right of the line through the centres.

The word K1 =.. for a connecting circle **outside the smaller circle** is:



- for the **left** connecting circle: K1=16
- for the **right** connecting circle: K1=26

The word K1=.. for a connecting circle **surrounding the smaller circle** is:



- for the **left** connecting circle: K1=17
- for the **right** connecting circle: K1=27

Note: It depends on the programmed direction of movement (G2 and G3) on the three circles which default value for K1 = is used by the control.

The formats are:

start point from N1 is known

N1 G2/G3 I.. J.. R1=1/2
 N2 G2/G3 R.. K1=16/17 or K1=26/27
 N3 G2/G3 etc.

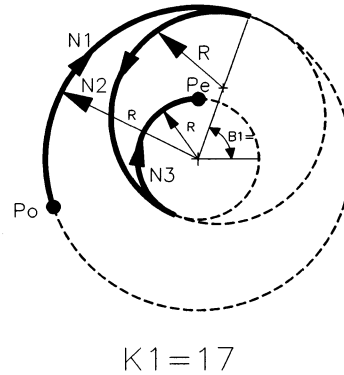
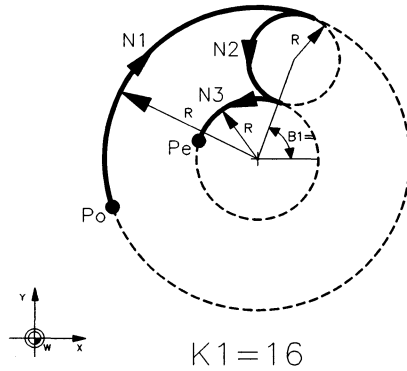
start point from N1 is not known

N1 G2/G3 I.. J.. R.. R1=1/2
 N2 G2/G3 R.. K1=16/17 or K1=26/27
 N3 G2/G3 etc.

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

CONNECTING CIRCLE WITH TWO CONCENTRIC CIRCLES

If both circles are concentric, the programming is the same as with one circle inside the other one, except that it is also necessary to program the angle ($B1=..$) which the line through the common centre point and the centre point of the connecting circle makes with the main axis.



NB8635

For both cases the following format is available:

start point from N1 and radius of the connecting circle are known

```
N1 G2/G3 I.. J.. R1=1/2
N2 G2/G3 R.. B1=.. K1=16/17
N3 G2/G3 I.. J..
```

start point from N1 and radius of the second circle are known

In this case the radius of the connecting circle is calculated by the control.

```
N1 G2/G3 I.. J..
N2 G2/G3 B1=.. K1=16/17
N3 G2/G3 I.. J.. X.. Y..
```

Refer to the corresponding section with continuous movements for the formats of block N1 and N3.

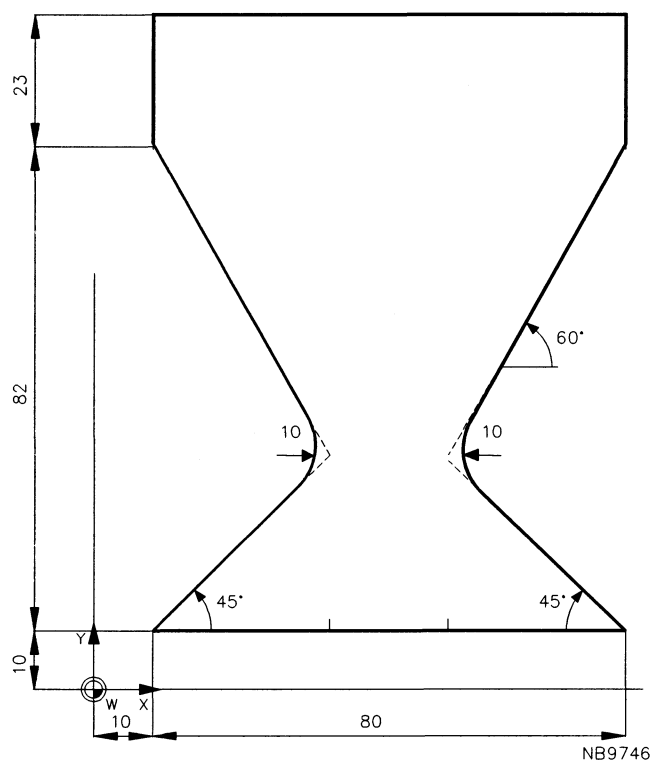
start point from N1 is not known

```
N1 G2/G3 I.. J.. R..
```

The other blocks from the mentioned cases remain the same.

Examples

EXAMPLE 1. Calculation of intersection point



```

N64001
N1 G54
N2 S1000 T1 M6 (RADIUS 2 mm)
N3 G0 X0 Y0 Z10 M3
N4 G1 Z-10 F500
N5 G43 Y10
N6 G42
N7 G64
N8 X90
N9 B1=135
N10 G2 R10
N11 G1 X90 Y92 B1=60
N12 Y115
N13 X10
N14 Y92
N15 B1=-60
N16 G2 R10
N17 G1 X10 Y10 B1=45
N18 G40
N19 G63
N20 X0 Y0
N21 G0 Z100
N22 M30
    
```

Explanation

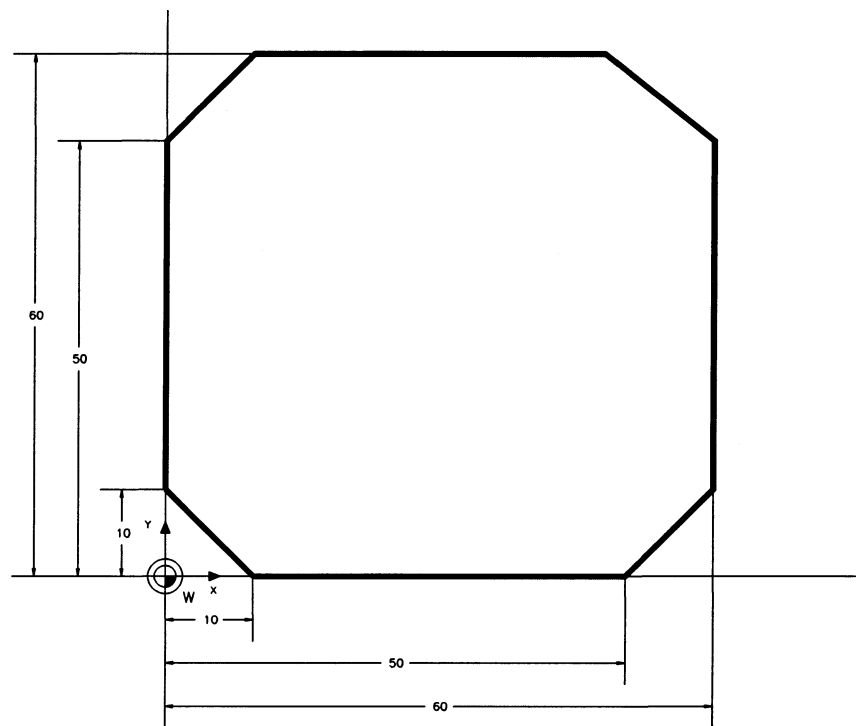
N1:	Set the program datum point
N2:	Load tool 1
N3:	Start the spindle and move tool to starting point
N4:	Feed tool to depth
N5:	Move tool T0 the contour
N6:	Set radius compensation RIGHT
N7:	Activate the geometric calculations
N8:	Move tool parallel to X-axis. Y-coordinate can be omitted.
N9:	A linear movement under an angle. The starting point of this line is known, so programming the angle is sufficient
N10:	The rounding between the intersecting lines of N9 and N11
N11:	A linear movement under an angle to an end point.
N12-N14:	Axis parallel movements.
N15-N17:	The linear movements under an angle including the rounding (N16) between these movements.
N18:	Cancel radius compensation
N19:	Cancel geometric calculations
N20:	Move the tool to a point free from the part
N21:	Retract the tool in the tool axis
N22:	End of the program

Notes:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

```
N801 G98 X0 Y0 Z0 I100 J130 K-10  
N802 G99 X10 Y10 Z0 I80 J105 K-10
```

EXAMPLE 2. Insertion of a chamfer between linear movements



NB6897

```

N64002
N1 G54
N2 S1000 T1 M6 (RADIUS 2 mm)
N3 G0 X-20 Y-20 Z10 M3
N4 G1 Z-10 F500
N5 G43 Y0
N6 G42
N7 G64
N8 X60
N9 I10
N10 Y60
N11 I10
N12 X0
N13 I10
N14 Y0
N15 I10
N16 X10
N17 G40
N18 G63
N19 Y-20
N20 G0 Z100
N21 M30
    
```

Explanation

N1 - N7:	Refer to the corresponding lines in the first example.
N8-N10-N12-N14:	Axis parallel movements
N9-N11-N13-N15:	Chamfer of N9 between the linear movements of N8 and N10, chamfer of N11 between N10 and N12 etc.
N16:	Last movement to define the position of the chamfer
N17:	Cancel radius compensation
N18:	Cancel geometric calculations
N19:	Move the tool to a point free from the part
N20:	Retract the tool in the tool axis
N21:	End of the program

Notes:

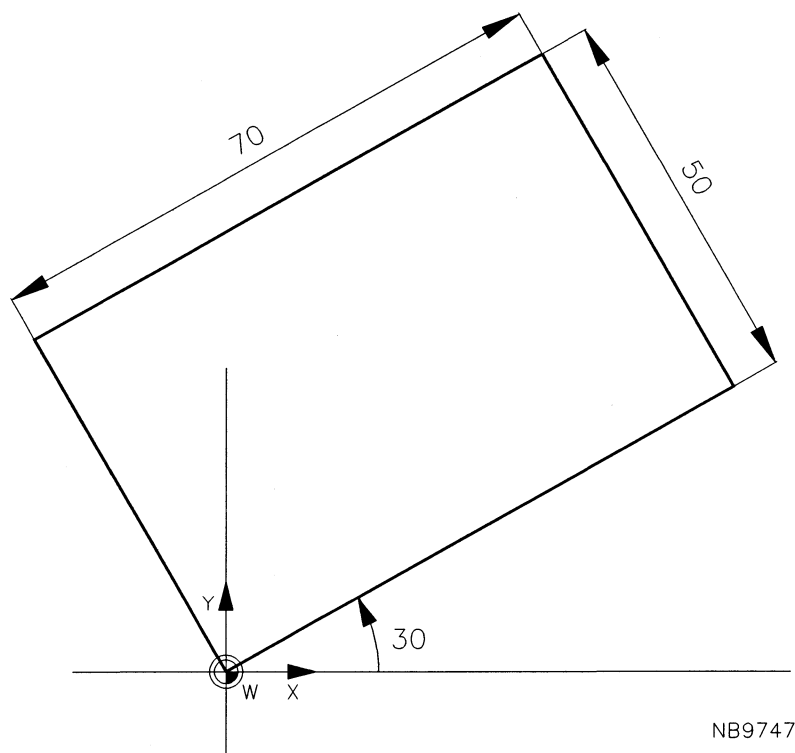
1. To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

N801	G98	X-10	Y-10	Z0	I80	J80	K-10
N802	G99	X-5	Y-5	Z0	I70	J70	K-10

2. If a rounding should be programmed in stead of the chamfer a few minor changes have to be made:

N9	G3	R10
N10	G1	Y60
N11	G3	R10
N12	G1	etc.

EXAMPLE 3. Parallel lines



NB9747

```

N64003
N1 G54
N2 S1000 T1 M6 (RADIUS 2 mm)
N3 G0 X-10 Y0 Z10 M3
N4 G1 Z-10 F500
N5 G43 X-4.33 Y-2.5
N6 G42
N7 G64
N8 X0 Y0 B1=30 I1=0
N9 X0 Y0 B1=120 I1=70
N10 X0 Y0 B1=-150 I1=50
N11 X0 Y0 B1=-60
N12 G40
N13 G63
N14 X-10 Y-10
N15 Z100
N16 M30
    
```


Explanation

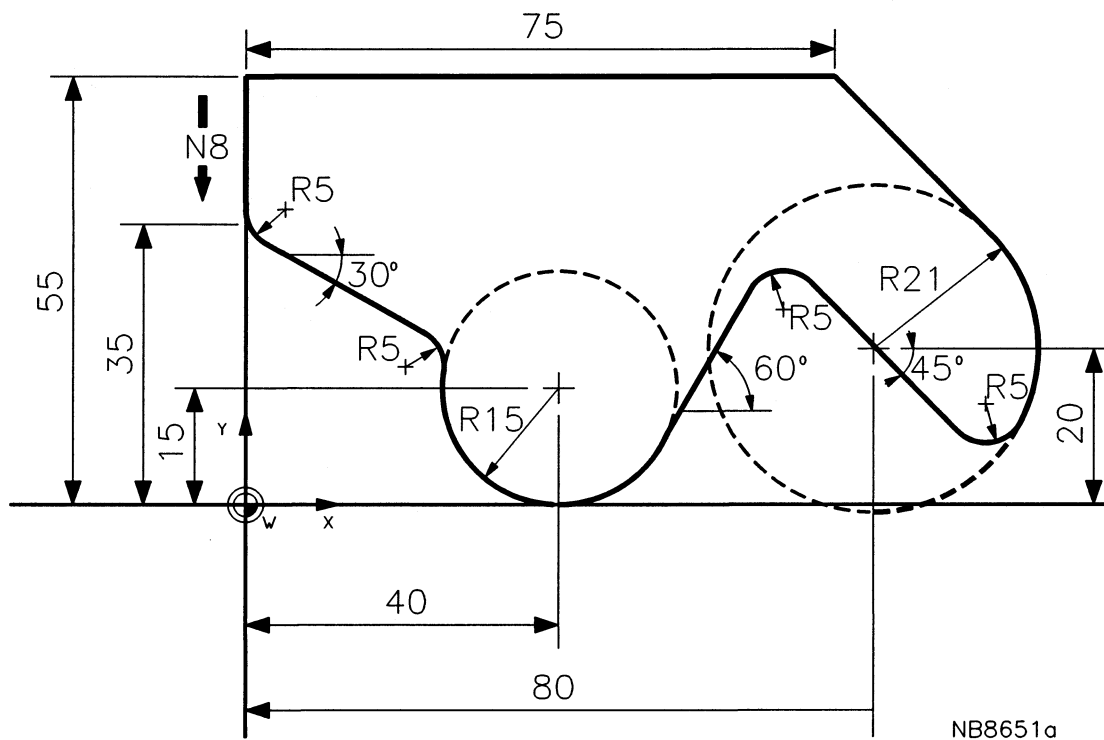
N1 - N7:	Refer to the corresponding lines in the first example.
N8:	Line defined by a support point (X0,Y0) and the angle with the X-axis
N9:	Line defined by a support point (X0, Y0), the angle with the X-axis and at a distance of 70 from the line through the origin.
N10:	Line defined by a support point (X0,Y0), the angle with the X-axis programmed in the direction of movement, and at a distance of 50 from the line through the origin.
N11:	Line defined by the endpoint (X0, Y0) and the angle with the X-axis programmed in the direction of movement.
N12:	Cancel radius compensation
N13:	Cancel geometric calculations
N14:	Move the tool to a point free from the part
N15:	Retract the tool in the tool axis
N16:	End of the program

Note:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

N801	G98	X-35	Y-15	Z0	I110	J105	K-10
N802	G99	X-30	Y-10	Z0	I120	J95	K-10

EXAMPLE 4. Line-to-circle and circle-to-line intersection



N64004
 N1 G54
 N2 S1000 T1 M6 (RADIUS 2 mm)
 N3 G0 X-5 Y60 Z10 M3
 N4 G1 Z-10 F500
 N5 G43 X0
 N6 G42
 N7 G64
 N8 B1=-90
 N9 G3 R5
 N10 G1 X0 Y35 I1=0 B1=-30 J1=2
 N11 G2 R5
 N12 G3 I40 J15 R15 R1=0
 N13 G1 B1=60
 N14 G2 R5
 N15 G1 X80 Y20 I1=0 B1=-45 J1=2
 N16 G3 R5
 N17 I80 J20 R21 R1=0
 N18 G1 X75 Y55
 N19 X-20
 N20 G40
 N21 G63
 N22 G0 Z100 M30

Explanation

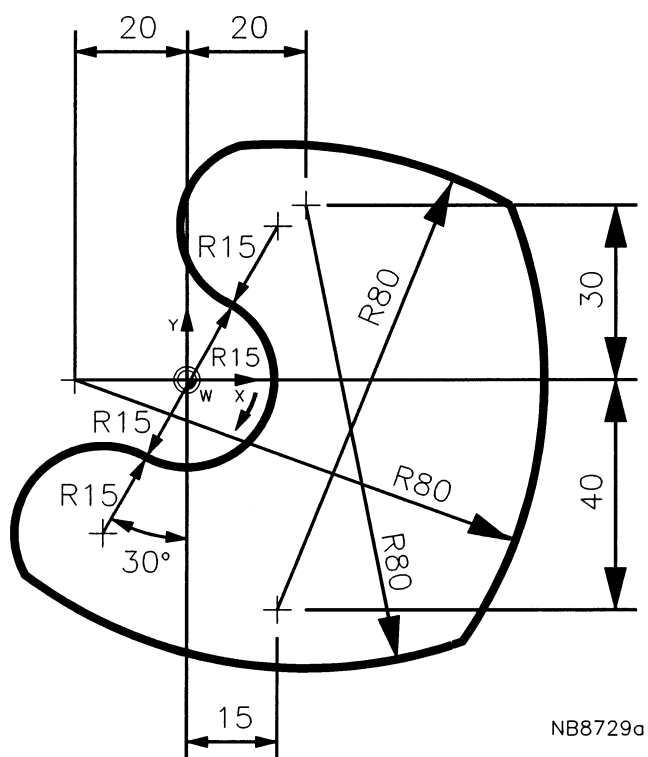
N1 - N7:	Refer to the corresponding lines in the first example.
N8:	Move tool downwards along the Y-axis
N9:	Make the rounding between the linear movements of N8 and N10
N10:	Move tool along the line. The starting point of this line is programmed as a support point, the angle is programmed in the direction of the movement and the right intersection point (J1=2) of the line and the circle of N12 should be used.
N11:	Make the rounding between the linear movement of N10 and the circular movement of N12
N12:	Follow the circle till the point of tangency between the circle and the linear movement of N13
N13:	A linear movement. The point of tangency is the known starting point.
N14:	A rounding between the linear movements of N13 and N15
N15:	A linear movement through the centre of the circle of N17. The centre point is used as a support point of the line. The intersection point in the direction of movement should be used (J1=2)
N16:	Make the rounding between the linear movement of N15 and the circular movement of N17
N17:	Follow the circle till the point of tangency between the circle and the linear movement of N18
N18:	A linear movement to the programmed end point of the line
N19:	Move tool parallel to X-axis, until the tool is free from the part
N20:	Cancel radius compensation
N21:	Cancel geometric calculations
N22:	Retract the tool in the tool axis and end of the program

Note:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

N801	G98	X-10	Y-20	Z0	I140	J100	K-10
N802	G99	X0	Y-10	Z0	I120	J80	K-10

EXAMPLE 5. Circle-to-circle intersection



```

N64005
N1 G54
N2 S1000 T1 M6 (RADIUS 2 mm)
N3 G0 X0 Y0 Z10 M3
N4 G1 Z-10 F500
N5 G43 X15
N6 G42
N7 G64
N8 G2 R15 R1=0
N9 G3 R15 B3=-120 L3=30 J1=1
N10 I20 J30 R80 J1=1
N11 I-20 J0 R80 J1=1
N12 I15 J-40 R80 J1=1
N13 R15 B3=60 L3=30 R1=0
N14 G2 X15 Y0 R15
N15 G40
N16 G1 X0
N17 G63
N18 G0 Z100 M30
    
```

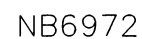
Explanation

- N1 - N7: Refer to the corresponding lines in the first example.
- N8: Move along the circle till the point of tangency with the next circle. The starting point of this circle is known from the previous blocks.
- N9: Move along the circle defined by its polar centre point coordinates (B3=, L3=..) and the radius. The left intersection point (J1=1) with the circle in N10 should be used.
- N10: Move along the circle defined by its cartesian centre point coordinates and the radius. The left intersection point (J1=1) with the circle in N11 should be used.
- N11 - N12: The same type of movement as N10.
- N13: Move along the circle till the point of tangency with the circle in N14. The circle is defined by its polar centre point coordinates (B3=, L3=..) and the radius.
- N14: Move along a circle programmed with end point and radius.
- N15: Cancel radius compensation
- N16: Cancel geometric calculations
- N17: Move the tool to a point free from the part
- N18: Retract the tool in the tool axis
- N19: End of the program

Notes:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

```
N801 G98 X-45 Y-60 Z0 I140 J140 K-10
N802 G99 X-40 Y-60 Z0 I125 J120 K-10
```



578

Explanation

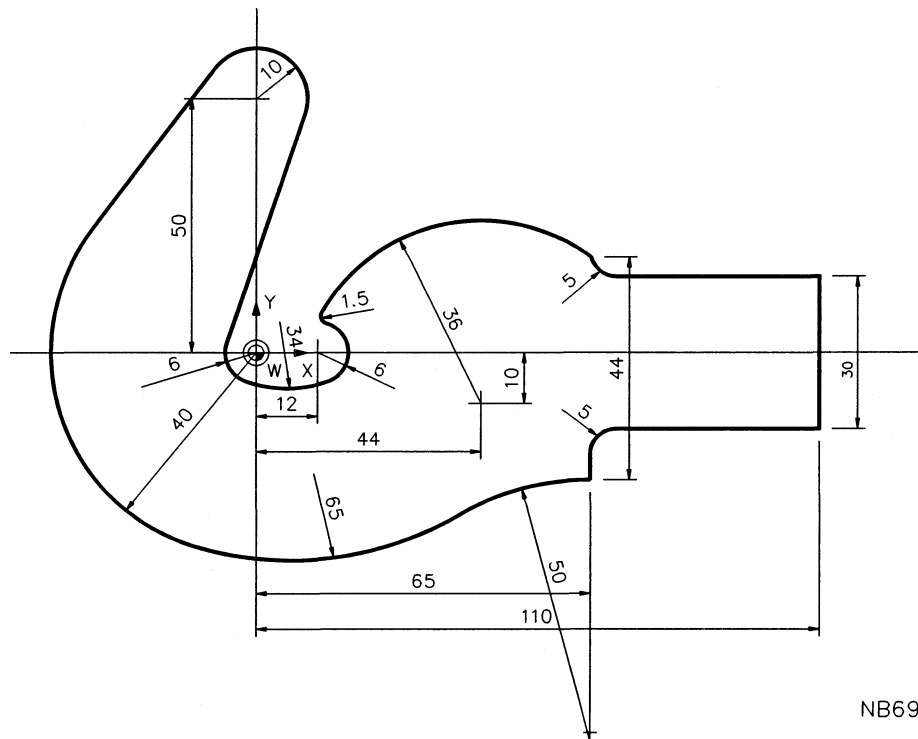
N1 - N7:	Refer to the corresponding lines in the first example.
N8:	Move along the X-axis. The starting point of the line is known from the previous blocks.
N9:	A surrounding connecting circle between the line of N8 and the circle of N10. The line does not meet the circle.
N10:	A circular movement till the point of tangency with the linear movement of N11. The circle is programmed with the cartesian centre point coordinates and the radius.
N11:	A linear movement. The starting point of the line is the point of tangency with N10. The line does not meet the circle of N13.
N12:	A connecting circle between the line of N11 and the circle of N13.
N13:	A circular movement from the point of tangency with the circle of N12 till the point of tangency with the circle of N14.
N14:	Similar to N12.
N15:	A linear movement to the point of tangency with the circle from N16.
N16:	A circular movement till the point of tangency with the surrounding connecting circle of N17
N17:	The surrounding connecting circle.
N18:	A linear movement along the X-axis to three programmed end point
N19:	Cancel radius compensation
N20:	Cancel geometric calculations
N21:	Retract the tool in the tool axis
N22:	End of the program

Note:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

```
N801 G98 X-90 Y-15 Z0 I180 J90 K-10
N802 G99 X-70 Y-5 Z0 I140 J60 K-10
```

EXAMPLE 7. Connecting circle between circles



NB6928a

```

N64007
N1 G54
N2 S1000 T1 M6 (RADIUS 2 mm)
N3 G0 X120 Y-35 Z10 M3
N4 G1 Z-10 F500
N5 G43 Y-15
N6 G41
N7 G64
N8 B1=180
N9 G3 R5
N10 G1 I65 Y-22 I1=0 B1=-90 J1=1
N11 G3 I65 J-72 R50
N12 G2 R65
N13 G2 I0 J0 R40 R1=0
N14 G1 R1=0
N15 G2 I0 J50 R10 R1=0
N16 G1 R1=0
N17 G3 I0 J0 R6
N18 G3 R34
N19 G3 I12 J0 R6 J1=1
N20 G2 R1.5
N21 G2 I44 J-10 R36 R1=0
N22 G1 X65 Y22
N23 B1=-90
N24 G3 R5
N25 G1 X110 Y15 B1=0
N26 Y-40
N27 G40
N28 G63
N29 X120
N30 G0 Z100 M30
    
```

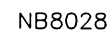

Explanation

N1 - N7:	Refer to the corresponding lines in the first example. The radius compensation is set to be LEFT.
N8:	Move parallel to the X-axis. The starting point is known from the previous blocks.
N9:	A rounding between the linear movements of N8 and N10
N10:	A downwards linear movement parallel to the Y-axis till the left intersection point of the circle from N11
N11:	A circular movement to the starting point of the connecting circle of N12
N12:	Connecting circle outside the circle of N11 and surrounding the circle of N13
N13:	Circular movement till the point of tangency with the line of the next block
N14:	Common tangent line between the circles of N13 and N15
N15:	Circular movement to the point of tangency with the line of the next block
N16:	Common tangent line between the circles of N15 and N17
N17:	Circular movement to the point of tangency with the connecting circle of the next block
N18:	Connecting circle which surrounds the circles from N17 and N19
N19:	Circular movement to the point of tangency with the rounding of the next block. The circle intersects the circle of block N21
N20:	The rounding between the two intersecting circles of N19 and N21
N21:	Circular movement to the point of tangency with the line of the next block.
N22:	Linear movement programmed with its end point
N23:	A downwards linear movement parallel to the Y-axis
N24:	A rounding between the linear movements of N23 and N25
N25:	Linear movement parallel to the X-axis programmed with end point and angle
N26:	Downward linear movement parallel to Y-axis
N27:	Cancel radius compensation
N28:	Cancel geometric calculations
N29:	Move the tool to a point free from the part
N30:	Retract the tool in the tool axis and end of the program

Note:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

```
N801 G98 X-50 Y-50 Z0 I170 J120 K-10
N802 G99 X-45 Y-45 Z0 I160 J110 K-10
```



582

Explanation

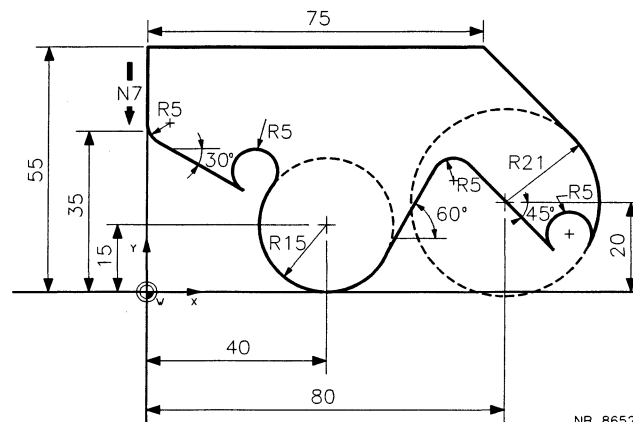
N1 - N7:	Refer to the corresponding lines in the first example. The radius compensation is set to be LEFT.
N8:	Linear movement. The line is tangent to the circle of N9. The starting point of the line is known from the previous blocks.
N9:	Circular movement to the point of tangency with the connecting circle of the next block
N10:	Connecting circle between the circles of N9 and N11. The greater arc is required. In this case the indicator K1=22 should be programmed. If this indicator is omitted the shorter arc (K1 =12) is automatically chosen by the control.
N11:	Circular movement to the point of tangency with the common tangent line of N12.
N12:	Common tangent line between the circles of N11 and N13.
N13:	Circular movement to the point of tangency with the connecting circle of the next block
N14:	Connecting circle on the outside of the circles from N13 and N15
N15:	Circular movement to the point of tangency with the line of N16.
N16:	Linear movement to the programmed end point.
N17:	Linear movement parallel to the X-axis
N18:	Cancel radius compensation
N19:	Cancel geometric calculations
N20:	Retract the tool in the tool axis
N21:	End of the program

Note:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

```
N801 G98 X-10 Y-10 Z0 I120 J80 K-10
N802 G99 X-5 Y-5 Z0 I110 J70 K-10
```

Example 9. Non-continuous movement between line and circle



```

N64009
N1 G54
N2 S1000 T1 M6 (RADIUS 2 mm)
N3 G0 X-5 Y60 Z10 M3
N4 G1 Z-10 F500
N5 G43 X0
N6 G42
N7 G64
N8 B1=-90
N9 G3 R5
N10 G1 X0 Y35 I1=0 B1=-30 J1=2
N11 G2 R5 K1=4
N12 G3 I40 J15 R15 R1=0
N13 G1 B1=60
N14 G2 R5
N15 G1 X80 Y20 I1=0 B1=-45 J1=2
N16 G2 R5 K1=4
N17 G3 I80 J20 R21 R1=0
N18 G1 X75 Y55
N19 X-20
N20 G40
N21 G63
N22 G0 Z100 M30
    
```

Explanation

Compare this program with the one of example 4. The differences are:

- N11: The direction of rotation on the circle is changed and the indicator K1= 4 programmed to make the required circular movement.
- N16: The same changes as in block N11.

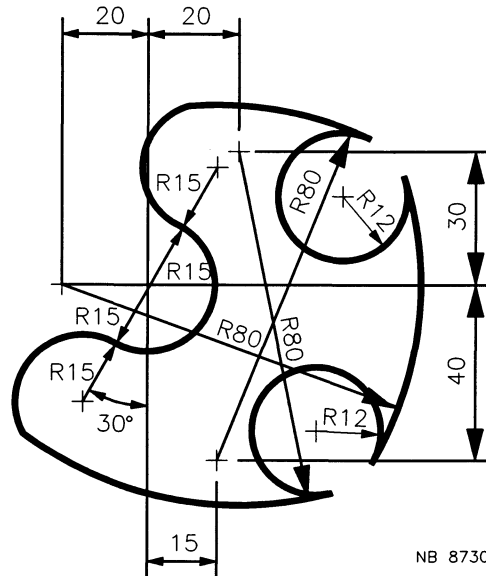
Note:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

```

N801 G98 X-10 Y-20 Z0 I140 J100 K-10
N802 G99 X0 Y-10 Z0 I120 J80 K-10
    
```

Example 10. Non-continuous movement between two circles



```

N64010
N1 G54
N2 S1000 T1 M6 (RADIUS 2 mm)
N3 G0 X0 Y0 Z10 M3
N4 G1 Z-10 F500
N5 G43 X15
N6 G42
N7 G64
N8 G2 R15 R1=0
N9 G3 R15 B3=-120 L3=30 J1=1
N10 I20 J30 R80 J1=1
N11 G2 R12 K1=4
N12 G3 I-20 J0 R80 J1=1
N13 G2 R12 K1=4
N14 G3 I15 J-40 R80 J1=1
N15 R15 B3=60 L3=30 R1=0
N16 G2 X15 Y0 R15
N17 G40
N18 G1 X0
N19 G63
N20 G0 Z100
N21 M30

```

Explanation

Compare this program with the one of example 5. The differences are the roundings of the blocks N11 and N13. The indicator K1 =4 is programmed to make the required circular movement between the intersecting circles.

Notes:

To get a graphical presentation of this program in one of the graphic modes of the control, you can add the following lines after N1:

```

N801 G98 X-45 Y-60 Z0 I140 J140 K-10
N802 G99 X-40 Y-60 Z0 I125 J120 K-10

```


107. Graphical support

The CNC PILOT provides the following types of graphical support:

- synchron graphic

the actual tool movements on the machine are simultaneously displayed on the control.

- graphical simulation:

the visualization of the programmed tool movements on the display of the control.

- graphical programming support

To get a graphical presentation on the display of the control the proper display mode must be chosen on the control, but also the part program should be enlarged with some special G-functions:

- G195 for defining the 3D window, thus a bounded area on the display of the control in which the graphical simulation takes place;
- G196 to G199 for defining the inner and outer contour of the blank part and the machine parts.

1. SYNCHRON GRAPHIC

With synchron graphic a 3D view of the cutter path (= path of the tool tip) is displayed on the control simultaneously with the actual tool movements on the machine tool. Feed movements and rapid traverse movements are displayed with different colours.

The graphic window defined with the G195 function is used to set the borders on the screen. There is no scale available to make any measurement.

The synchron graphics is available in the mode OPERATE of the control.

2. GRAPHICAL SIMULATION

Graphical simulations allow programmed tool movements to be visualized on the display of the control.

The visualization includes:

- a wire plot graphic showing the contour, the tool movements, the cycle end points, the machine parts and the tool shape.
- an erase graphic showing the machining of the workpiece and if required indicating a collision between tool and workpiece with rapid traverse movements or machine parts,
- a workpiece presentation showing the machined workpiece and the intersections.

Graphical simulations of a part program are available in the mode OPERATE of the control.

Note: Rotary axes are not simulated.

WIRE PLOT GRAPHIC

In wire plot graphic the outer and inner contour of the blank, the machine parts and the tool shape are shown. The tool movements along the part are generated in either SINGLE BLOCK or AUTOMATIC mode. The functions BLOCK DELETE and OPTIONAL STOP can also be executed during the execution of the program.

The axis cross can be shown to measure the part dimensions and the position of the tool.

A mill is represented as a circle. Its size is picked up from the tool memory.

In all plot modes except the 2.5 D plot, a picture of the tool holder and the tool defined with the G-word from the tool memory can be displayed in a separate window.

The programmed model of the blank, the machine parts, the tool shape picture, the programmed contour, the feed and the rapid traverse movements are all displayed in different colours.

Collisions during rapid traverse movements between the tool and the workpiece or collisions with machine parts can be seen.

The end contour of a fixed milling cycle is shown as a programmed contour.

With the universal pocket cycle the programmed contour of the pocket and the islands are shown when the function G200 is processed. Then the macros are generated and the pocket milling simulated.

A zoom function is available to expand or reduce the window for viewing the part

2D plot

With a 2D plot both the contour and the cutter path in the active main plane can be displayed. In this way the programmed contour and the cutter path along that contour can be checked.

2.5D plot

With a 2.5D plot the tool movements in the active main plane and in the projections are shown. With a machine constant setting either european or american projection can be chosen.

In a small extra window a 3D plot of the part can be shown.

In this plot mode the contour and the cutter path along that contour can be checked, but also the movements in the tool axis.

In the 2D and 2.5D plot mode a cross hair function is available to check the part dimensions. The position of the cross is shown in the top window.

3D plot and Perspective view

With the 3D plot and the perspective view the tool movements are shown in a spatial representation of the workpiece.

With the 3D plot parallel lines remain parallel and with the perspective view they meet at a particular horizon.

In both plot modes the part can be rotated about the three main axes to show the part under different angles.

ERASE GRAPHIC

With erase graphics a solid model of the part is displayed in either a 2D or a 2.5D view. In this model the machining of the part is shown and if required a collision is detected between the part and the tool during rapid traverse movements or with the machine parts, if programmed in the graphical model.

Different colours are also used for different depths of operation.

A zoom function is available to expand or reduce the window for viewing the part

An axis cross is not available. Measurements on the display can be executed in either the wire plot graphic or the work piece presentation.

COLLISION DETECTION WITH ERASE GRAPHIC

The programmed contours of the machine parts are not used for collision detection, but instead the smallest possible rectangle is calculated which encloses the whole of the machine part(s). When a tool is detected crossing into this rectangle a collision warning is generated.

The softkey 'COLL DETECT' must be activated for a collision message to be generated during a graphical simulation. If this softkey is not activated, the graphic tool will just draw over the machine part.

WORKPIECE PRESENTATION

With work piece presentation the total program is executed by the control in a fast way, then the part and its intersections are displayed. The user can select the intersection and move the cross hair to the point he wants to check. The part is then displayed in all three intersections.

3. GRAPHICAL PROGRAMMING SUPPORT

With programming via the control, thus entering a program via the control, three types of support are available:

- a. **Static pictures** explaining the addresses belonging to a selected G-function. These pictures are available with "assisted entry" and "dialogue entry" of the control.

- b. **Dynamical graphic.**

With "assisted entry" and "dialogue entry" of the control there is a 2D wire plot graphic available for showing the programmed contour and the cutter path of the tool moving around the contour.

As soon as a few blocks are entered the dynamical graphics can be used to show the effect of the programmed blocks. So changes can be made immediately.

The window defined in a G195 block is used in this graphic mode.

- c. **Interactive Contour Programming (ICP)**

The dynamical graphic within ICP allows to see on the display each program block. ICP does not only show the blocks which are ready, but also indicates the possible choices, such as the intersection point of a line and a circle or two circles.

If elements cannot be drawn, ICP shows in a special window up to three elements which are programmed, but cannot be used up to now. Once the control can calculate the end points of the blocks, all elements are shown on the graphic.

Depending on the ICP-contour size, a special window is used in the ICP mode for the geometry section of the program.

108. Tool memory

The tool memory of the control can be used to store tool dimensions and other tool related parameters. This memory can be used in a FMS environment, but also outside such an environment.

In this section a general description of the parameters in the tool memory is given.

Refer also to:

- the machine tool builder's documentation to see which monitoring devices are activated on your machine tool,
- the user manual for entering the tool data into the memory.

RANDOM ACCESS TOOL MAGAZINE

When a tool magazine can be filled at random, a table containing per tool its place in the magazine and the corresponding tool identification number should be stored in the tool memory of the control before the first run of the program.

At a tool change (M6) the programmed tool is picked up from the magazine and the used tool put back at the empty place of the loaded tool. The table of tool places is automatically updated by the control.

PLACE OF TOOL IN MAGAZINE (P)

The three digit P-word in the tool memory is used for indicating the place of the tool in the magazine, where P1 corresponds to place 1, P2 to place 2, etc.

The actual number of places in the magazine is stored as a Machine Constant.

OVERSIZED TOOLS (S)

With the S-word in the tool memory is indicated if a tool occupies one place (S0) or is oversized (S1). In the latter case the tool occupies three places in the magazine, one place in which the tool is stored and two empty places at the left and right of the tool.

After using such a tool it should be put back at the same place in the magazine.

TOOL IDENTIFICATION NUMBER (T)

The tool number in the tool memory is used to identify the tool. It is entered with the address T and a value with eight digits before and two digits behind the decimal point. The eight digits are reserved for the tool identification number.

The two digits behind the decimal point specify a spare tool related to the tool.

SPARE TOOLS

A spare tool can replace the tool after its working life has ended or the lowest power level for this tool in the cutting force monitor, if available, is exceeded.

The spare tool is a two digit number placed behind the decimal point of the tool identification number. Therefore, a maximum of 99 spare tools can be assigned to the same tool.

TOOL DIMENSIONS (L, L1=, L2=, R, R1=, R2=, C, C1=, C2=)

A tool can have a length (L-word) and two extra length values (L1 = and L2=), a radius (R-word) and two extra radius values (R1= and R2=) and a corner radius (C-word) and two extra values (C1= and C2=).

Refer to the T-word EXTRA TOOL OFFSETS (T2=) for activating the extra tool offsets.

Note: The length value is used with the length compensation, the radius value with radius compensation (G41 to G44 or G141), the corner radius is used with 3D radius correction (G141).

GRAPHIC PARAMETER (G)

This parameter is used to define the tool shape. The available shapes are displayed when entering tool data into the tool memory.

The tool shape is used with the graphical simulation of a part program.

TOOL TYPE (Q3=)

The tool type parameter in the tool memory (Q3=) has to be the same as the Q3= parameter in the technology table.

If the material, type of operation and tool number are entered, when asking for a technology proposal, this parameter is automatically picked up from the tool memory.

NUMBER OF CUTTING EDGES (Q4=)

This parameter indicates the number of cutting edges of a mill. If a technology proposal is asked for during entering a program via the control, this parameter is picked up from the tool memory, provided that the tool number and Q3= parameter are already entered.

ENABLING/DISABLING A TOOL (E)

The E-word in the tool memory indicates if the tool can be used or not.

E-1: tool is disabled, cannot be used. This parameter is set by the control if the tool life is ended or the lowest power level of the cutting force monitor is exceeded.

E0: tool can be used, but is not measured;

E1: tool can be used and is measured.

TOOL LIFE MONITORING (M, M1=, M2=)

With the word M2= in the tool memory is indicated that the tool life of a specific tool should be monitored by the CNC.

M2=0: no tool life monitoring for the specific tool

M2=1: tool life should be monitored

The word M in the tool memory is available to assign a working life in minutes to a tool. The stored tool life ranges from 1 to 99999 minutes.

Every time the tool is used, the working tool life in the memory is reduced with the cutting time. With the word M1= in the tool memory the remaining tool life is displayed.

When the tool life has expired a warning message is displayed, so that the tool can be replaced and the E-word in the tool memory set to -1 to indicate that the tool is disabled.

When the same tool is used again, program execution is either interrupted or a spare tool loaded (depending on the machine tool configuration).

A Machine Constant must be set to indicate that tool life monitoring for the tools is required.

TOOL BREAKAGE MONITOR (B, B1=)

With an external device mounted on the machine tool, the tool length is measured when the tool is loaded into the spindle and again when it is put back into the magazine. If the difference between the two measurements is greater than a tolerance value, an error message is displayed and the tool disabled.

The tolerance value is entered into the tool memory with the B-word.

With the word B1= in the tool memory is indicated that the tool breakage of a specific tool should be monitored by the CNC.

B1=0: no tool breakage monitoring for the specific tool

B1=1: tool breakage should be monitored

A Machine Constant must be set to indicate that tool breakage monitoring for the tools is required.

CUTTING FORCE MONITOR

With an external device mounted on the machine tool, the cutting force being applied to a tool, can be monitored by constantly measuring the power consumption of the spindle drive. When a power overload condition is detected, appropriate actions will be taken to prevent the workpiece or tool from being damaged.

Note

1. Cutting force monitoring is usually used with heavy cutting, generally with tools of >10 mm diameter.
2. Refer to the description of the T-word, Notes and Usage CUTTING FORCE MONITOR for activating and disabling the cutting force monitor in a program.
3. A Machine Constant must be set to indicate that cutting force monitoring for the tools is required.

CUTTING FORCE LEVELS

A group of three cutting force levels can be chosen for a tool. These levels are determined by such factors as the type of tool, the workpiece material and the type of machining operation to be performed. Up to 99 groups of power levels can be stored in the monitor's memory.

The monitoring system generates output signals to indicate which power level is exceeded by the cutting force.

LOWEST POWER LEVEL EXCEEDED

When the cutting force exceeds the lowest power level, this indicates that the tool is worn out. The CNC generates a warning message but continues using the tool until the next tool change or another power level is exceeded. The E-word in the tool memory is set -1 to indicate that the tool is disabled.

If the tool is used again an error message is generated or, if available, an assigned spare tool loaded.

MEDIUM POWER LEVEL EXCEEDED

When the cutting force exceeds the medium power level, the CNC responds to this message with an immediate feed/speed hold and an error message. The user has to intervene.

HIGHEST POWER LEVEL EXCEEDED

When the cutting force exceeds the highest power level, this indicates that the maximum permitted cutting force has been exceeded and an emergency stop is made. The user has to intervene.

109. Machine constants

General

MC 11:	define the plane to be set at start up of the control =17: XY-plane; =18: XZ-plane; =19: YZ-plane.
MC 19:	Type of projection with 2.5D graphic modes =0 european projection =1 american projection
MC 20:	Axes configuration with 2D and 2.5D graphic modes. G17 X-axis horizontal and Y-axis vertical G18 Z-axis horizontal and X-axis vertical G19 Y-axis horizontal and Z-axis vertical =0 zero point lower left =1 zero point upper left =3 zero point upper right =4 zero point lower right
MC 78:	M13 and M14 are recognized by the control.
MC 82:	Maximum number of defined points which can be stored in the control.
MC 83:	Maximum number of E-parameters which can be stored in the control.
MC 85:	Maximum number of partprograms and macros which can be stored in the control.
MC 135:	Feed limitation threshold value
MC 136:	The corner release distance
MC 137:	The maximum corner error

Programming

MC 707:	=70 control starts up in inch mode =71 control starts up in metric mode
MC 711:	Angle for intermediate circle
MC 714:	Type of scaling parameter and the axes involved. =0 scaling with a factor; main axes only =1 scaling with a percentage; main axes only =2 scaling with a factor; main axes and tool axis =3 scaling with a percentage; main axes and tool axis
MC 715:	Format of the scaling factor
MC 720:	Overlap parameter with pocket milling (cutting width)
MC 723:	Deceleration distance with G84
MC 724:	Dwell at the bottom with tapping G84

Machine constants

- MC 727: Minimum spindle speed with deceleration with G84
- MC 728: Pitch reduction with tapping G84
- MC 740: Maximum feed rate

Measuring with a touch trigger probe

- MC 840: Measuring probe connected
- MC 841: Type of measuring probe
=0 an inductive probe
=1 a remote signalling probe (eg. an infrared probe)
=4 a hard wired probe
- MC 842: Duration of air blow before measuring
- MC 843: Measuring feed rate
- MC 844: Pre-measuring distance
- MC 845: Post-measuring distance
- MC 846: Display resolution for the rotary axis with axis alignment
- MC 847: Width fixed measuring probe
- MC 848: Radius calibration ring
- MC 850: Collision protection measuring probe
=0 during all movements except the measuring movement
=1 during all feed movements except the measuring movement
=2 during all movements except the measuring movement and the retract after the measurement.
- MC 851: Delay time for the collision protection

Tool handling

- MC 27: Number of tools in the tool memory
- MC 28: Number of places in the tool magazine
- MC 29: Tool life monitoring activated
- MC 31: Cutting force monitoring activated
- MC 32: Tool breakage monitoring activated

External program selection

- MC 42: =0 no external program call
 =1 a fixed assignment of the program name
 =2 a variable assignment of the program name
- MC 43: With external program call the number of variable assignments which can be stored in the control.
- MC 44: =0 G14 and G29 are executed as described in the manual. The execution is not influenced externally.
 =1 G14 and G29 are executed as described in the manual but they can be influenced externally.

Communication

- MC 771: =0 no check on block numbers
 =1 check on same block numbers
- MC 772: =0 no syntax check on the entered block
 =1 a validation check on the entered block
- MC 773: =0 Blocknumbers < 9000
 =1 Blocknumbers >= 9000

110. External program selection

When automatic workpiece changers (eg. pallet stations) are used, it is necessary to activate and to execute a new part program without interference of the user.

The external part program selection allows:

- to assign to each part program a special identification;
- to store the identification of the next program during the execution of a part program;
- to activate the requested part program after finishing the running program (at M30);
- to execute the program after an external start command.

Two possibilities are available for the external selection of a part program:

1. FIXED ASSIGNMENT.

The last three digits of the identification number of the part program can be entered into the control via the interface of the machine tool.

In the part program the relation between the mounting zero point C and the machine zero point M_i can be established with one of the functions for the zero offsets (G54 - G59, G54I[nr.]) and the preset values (G52).

2. VARIABLE ASSIGNMENT.

A special three-digit identification number is assigned to a part program. These three digits can be entered into the control via the interface of the machine tool.

There is a special memory in the control, the external program call memory, in which are stored:

- the special identification number (E-word). The E-word ranges from 0 to 999, so 1000 programs can be assigned. The E-words are stored in the memory in increasing order. A machine constant determines how many assignments can be stored in this memory;
- the program identification (N-word) of the part program which is activated when the E-word is selected;
- the pallet offsets belonging to the pallet used with the workpiece.

PALLET OFFSETS WITH VARIABLE ASSIGNMENT



NB8613

The pallet offsets are related to the secondary machine zero point M_i . These offsets define the mounting zero point C.

After the external activation of a program the pallet offsets are loaded into the zero offset memory under G52 and this function is automatically activated.

Because G52 is automatically activated, it is advisable to end each part program with G51. This method prevents the pallet offsets of a previous program to be used in the program that follows.

Note:

Two machine constants must be set:

- one for activating the external program call function and to indicate what type of assignment is used.
- one for stating how many assignments can be stored in the memory of the control.

111. The V330 improvements over it's predecessors

Added:

- Calling actual axes-positions values (**G326**)

Modified functions:

- Programs bigger then 100 Kbyte will be executed by the program call (**G23**) as CAD-MODE programs. The programs will be not store in the work memory.

- The processing of measuring results (**50**) is extended with an aligning function for special machines, by which the C-axis can be turned 90 degrees.

- A movement in **G74** will not be done, if the machine positions (MC3145--MC3154) are zero.

- Read tool data and zero offset (**G149**) is extended.
 - The remaining tool life time (T3) can be read out.
 - The read out of the actual positions of the axes is added.
 - The tool status (E) is extended.

- Changing tool data and zero offset (**G150**) is extended.
 - The remaining tool life time (T3) can be changed.
 - The tool status (E) is extended.

- Graphic contour description for drawing von contours (**G199 B3**) is added.

- The calculation for the universal pocketcycli (**200 and G202**) will be ended by G202 or M30. After an other G200 the calculation will be started again.

112. Application notes

112.1 Reaming cycle with separate outfeed

Introduction

The movements in the reaming cycle G85 of the Heidenhain control are executed with the programmed feedrate into the hole and with the same feed out of it.

To save machining time an extension of this cycle is needed, so that the tool moves with feedrate into the hole and with a faster feed out of it.

Parameters with this cycle

E60: Code for the plane

=17 for XY-plane

=18 for XZ-plane

=19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

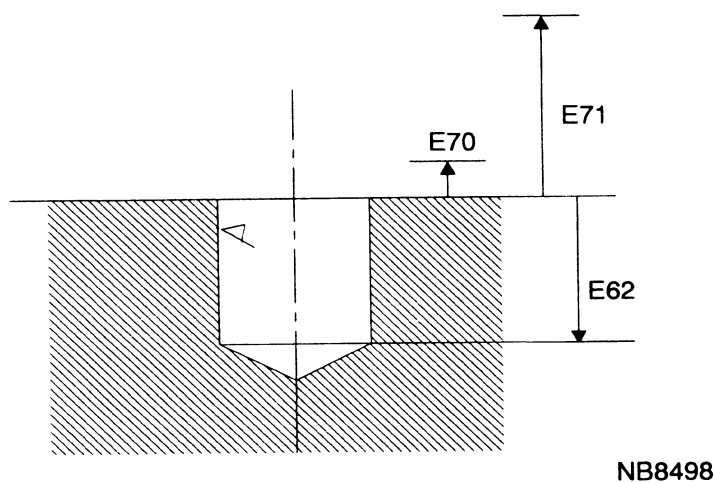


Figure AN001.-1. Meaning of the parameters

Application notes

E50: X-coordinate of the position for the cycle
E51: Y-coordinate of the position for the cycle
E52: Z-coordinate of the position for the cycle

E50, E51 and E52 are absolute coordinates.

E62: depth

E70: clearance distance
E71: the retract distance.
E72: the dwelltime (in seconds) at the bottom
E73: feedrate of the operation
E74: outfeed

Remarks:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. With all distances the sign should be considered.
+: in positive direction of the axis.
-: in negative direction of the axis.
3. The feed is:
with G94 active in mm/min (or inches/min)
with G95 active in mm/rev (or inches/rev).
4. All parameters must be defined at the macro call.

Local parameters: E90 for the unconditional jump
E91 used with the relational expression
E92 to E99 different meanings

Instruction for using the cycle

At the moment of the call the part program must be in absolute dimension mode (G90-active).

The macro

N99085 (REAMING CYCLE WITH OUTFEED)

N1 E90=1 E95=E52 E99=20

N2 G29 E91=E60=17 E91 N=6

N3 E95=E51 E99=14

N4 G29 E91=E60=18 E91 N=6

N5 E95=E50 E99=8

N6 E92=E95+E70 E93=E95+E62 E94=E95+E71

N7 G29 E90 K0 N=E99

N8 G0 X=E92

(CYCLE IN X-AXIS)

N9 G1 X=E93 F=E73

N10 G4 X=E72

N11 G1 X=E92 F=E74

N12 G0 X=E94

N13 G29 E90 K0 N=25

N14 G0 Y=E92

(CYCLE IN Y-AXIS)

N15 G1 Y=E93 F=E73

N16 G4 X=E72

N17 G1 Y=E92 F=E74

N18 G0 Y=E94

N19 G29 E90 K0 N=25

N20 G0 Z=E92

(CYCLE IN Z-AXIS)

N21 G1 Z=E93 F=E73

N22 G4 X=E72

N23 G1 Z=E92 F=E74

N24 G0 Z=E94

N25 F=E73

Example of the use

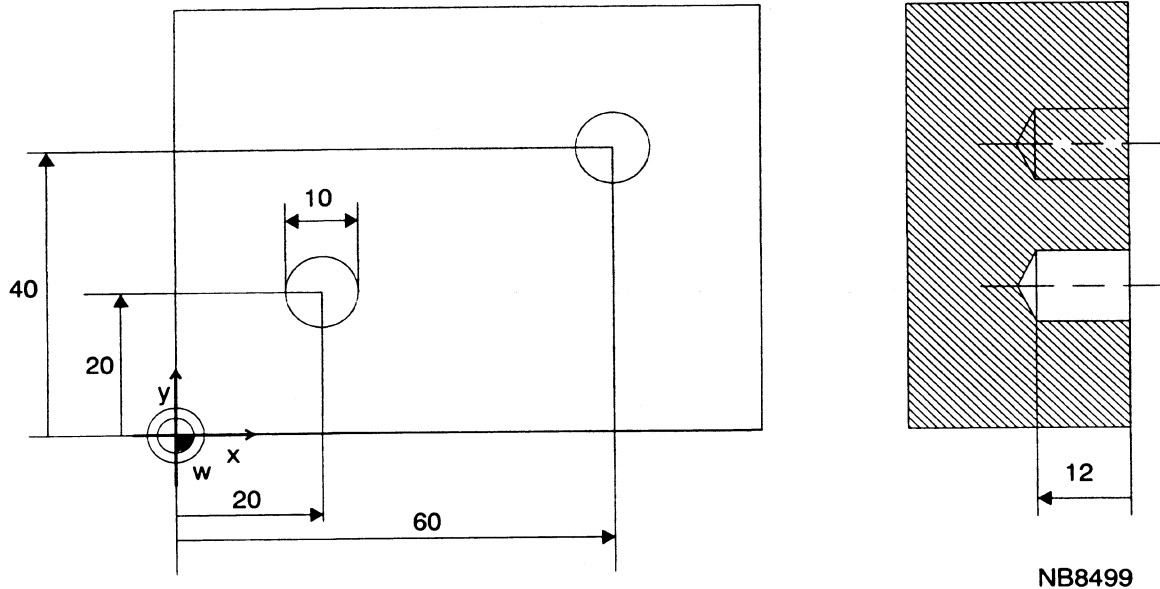


Figure AN001.-2 Two holes to be reamed

The two holes from figure AN001.-2 are drilled and reamed. In the part program the fixed cycle for drilling and the macro for reaming (N99085) are used. The part surface is the plane Z=0.

The part program could be:

```

N99185      (EXAMPLE OF REAMING MACRO)
N1 G54 E60=17
N2 T1 M6 (LOAD THE DRILL)
N3 G81 Y2 Z15 F200 S500      M3
N4 G79 X20 Y20 Z0
N5 G79 X60 Y40
N6 E62=-12 E70=2 E71=0 E72=0 E73=100 E74=500
N7 E50=20 E51=20 E52=0 S600 T2      M6      (LOAD THE REAMER)
N8 G22 N=99085
N9 E50=60 E51=40
N10 G22 N=99085
N11 M30
  
```

112.2 Boring cycle without dragline

Introduction

In the boring cycle G86 of the Groundage control the spindle is stopped and with the retract of the tool a dragline is made on the wall of the hole.

With the function "oriented spindle stop" (M19) the spindle is stopped in a well defined position. By positioning the tool out of the centre of the hole the dragline on the wall during the retract can be avoided. With the aid of a macro the boring is extended in this way.

Parameters with this cycle

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

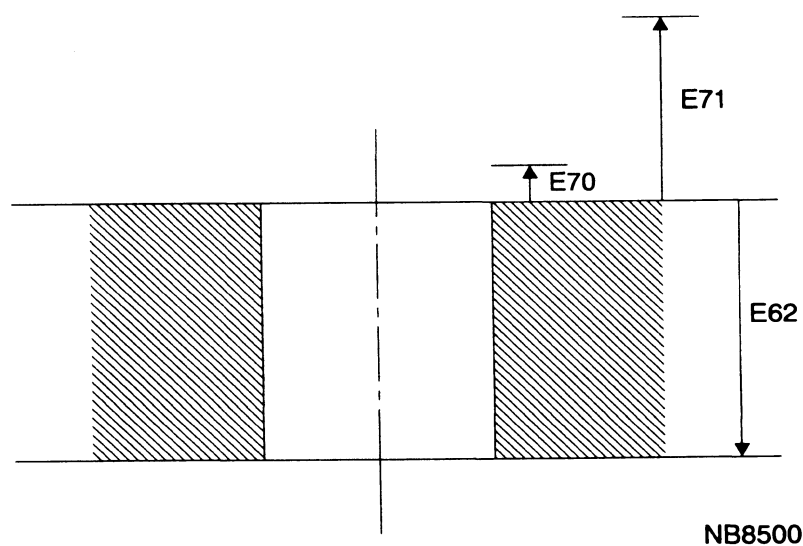


Figure AN002.-1. Meaning of the parameters

Application notes

E50: X-coordinate of the position for the cycle
E51: Y-coordinate of the position for the cycle
E52: Z-coordinate of the position for the cycle

E50, E51 and E52 are absolute coordinates.

E62: depth

E70: clearance distance
E71: the retract distance.
E72: the dwelltime (in seconds) at the bottom
E73: feedrate of the operation

E75: spindle speed in rev/min
E76: direction of spindle rotation
=3 for clockwise (M3)
=4 for counter clockwise (M4)

E77: displacement in X-axis (G17 or G18) or Y-axis (G19)
E78: displacement in Y-axis (G17) or Z-axis (G18 or G19)

Remarks:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. With all distances the sign should be considered.
+: in positive direction of the axis.
-: in negative direction of the axis.
3. The feed is:
with G94 active in mm/min (or inches/min)
with G95 active in mm/rev (or inches/rev).
4. All parameters must be defined at the macro call.

Local parameters: E90 for the unconditional jump
E91 used with the relational expression
E92 to E99 different meanings

Tool sequence

1. Move the tool in the toolaxis at rapid traverse to the point where the feed movement starts.
2. Move the tool with feed to depth.
3. An oriented spindle stop after the programmed dwell.
4. Move the tool out of the centre.
5. Retract the tool out of the hole.
6. Move the tool back to the centre.
7. Start the spindle again.

The macro

```

N99086      (BORING CYCLE WITHOUT DRAGLINE)
N1 E90=1 E95=E52 E97=E50 E98=E51 E99=23
N2 G29 E91=E60=17 E91 N=6
N3 E95=E51 E98=E52 E99=16
N4 G29 E91=E60=18 E91 N=6
N5 E95=E50 E97=E51 E99=9
N6 E92=E95+E70 E93=E95+E62 E94=E95+E71
N7 E97=E97+E77 E98=E98+E78
N8 G29 E90 K0 N=E99
N9 G0 X=E92 (CYCLE IN X-AXIS)
N10 G1 X=E93 F=E73
N11 G4 X=E72 M19
N12 G0 Y=E97 Z=E98
N13 X=E94
N14 Y=E51 Z=E52
N15 G29 E90 K0 N=29
N16 G0 Y=E92 (CYCLE IN Y-AXIS)
N17 G1 Y=E93 F=E73
N18 G4 X=E72 M19
N19 G0 X=E97 Z=E98
N20 Y=E94
N21 X=E50 Z=E52
N22 G29 E90 K0 N=29
N23 G0 Z=E92 (CYCLE IN Z-AXIS)
N24 G1 Z=E93 F=E73
N25 G4 X=E72 M19
N26 G0 X=E97 Y=E98
N27 Z=E94
N28 X=E50 Y=E51
N29 S=E75 M=E76

```

Instruction for using the boring cycle

1. The spindle position with the oriented stop must be known very well to the part programmer. He has to program a sign to the displacement values (E77 and E78), so the tool is properly positioned out of the hole.
If e.g. the tool stops at 12 o'clock, the displacement values can be E77=0 (X) and E78=-5 (Y). In this case the tool is positioned 5 mm out of the centre in the negative direction of the Y-axis.
2. If a dragline is allowed at a fixed position on the wall, the displacement values are 0, thus E77=0 and E78=0.
3. With M19 the spindle is stopped in a well defined position. If an angular attachment is used for a tool in an other direction then the Z-axis, it depends on the machine tool, if the tool is also stopped in a well defined position.
4. At the moment of the call the part program must be in absolute dimension mode (G90-active).

Example of the use

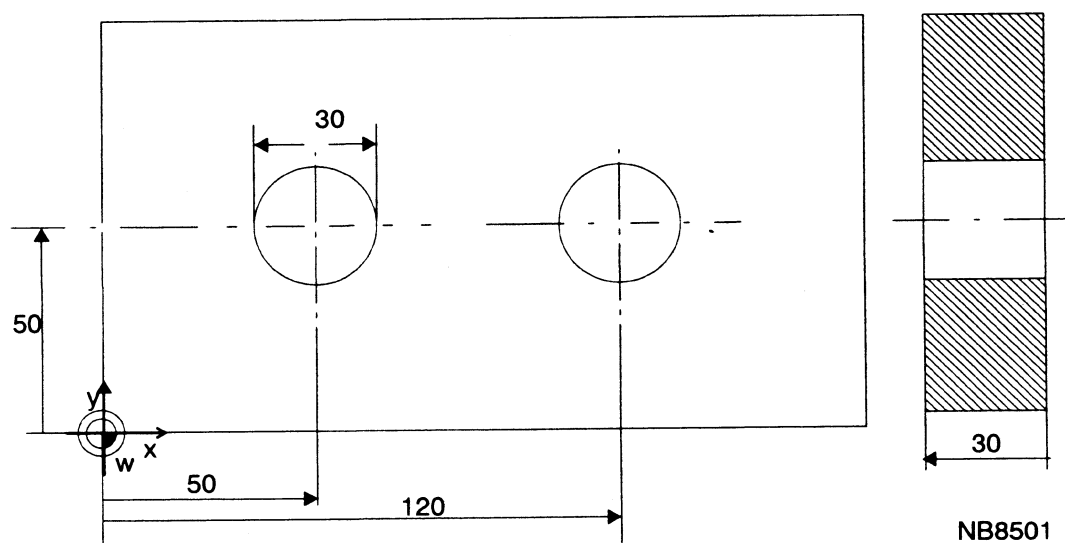


Fig. AN002.-2. Two holes to be bored without dragline

The two holes of figure AN002.-2. are bored. Macro N99086 is used for boring without a dragline. It is assumed that:

- the part surface is the plane $Z=0$
- the spindle stops at 12 o'clock.

The part program could be:

```

N99186 (EXAMPLE OF BORING WITHOUT DRAGLINE)
N1 G54 E60=17
N2 E50=50 E51=50 E52=15 T1 M6 (LOAD BORING BAR)
N3 E61=0 E62=-30 E70=5 E71=15 E72=0 E73=100
N4 E75=300 E76=3 E77=0 E78=-5 F100 S300 M3
N5 G22 N=99086
N6 E50=120
N7 G22 N=99086 M30
  
```

112.3 Back boring cycle

Introduction

Sometimes a back boring or a back facing operation has to be executed. A macro is programmed to perform this operation. The function "oriented spindle stop" (M19) is used to stop the spindle in a well defined position for entering and leaving the hole.

Parameters with this cycle

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

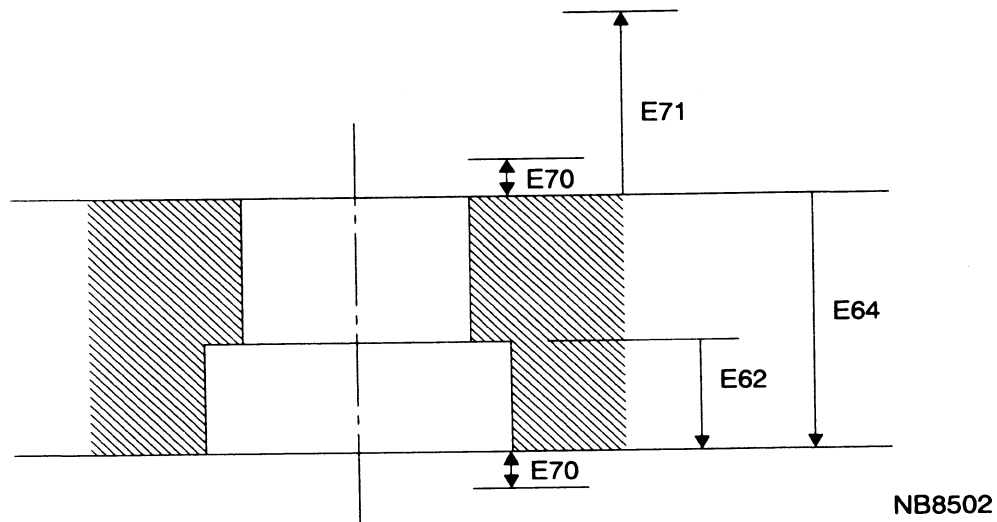


Figure AN003.-1. Meaning of the parameters

E50: X-coordinate of the position for the cycle
E51: Y-coordinate of the position for the cycle
E52: Z-coordinate of the position for the cycle

E50, E51 and E52 are absolute coordinates.

E62: depth (the same sign as E64)
E64: distance from the part surface to the surface where the back boring starts

E70: clearance distance

E72: the dwelltime (in seconds) at the bottom
E73: feedrate of the operation

E75: spindle speed in rev/min
E76: direction of spindle rotation
= 3 for clockwise (M3)
= 4 for counter clockwise (M4)

E77: displacement in X-axis (G17 or G18) or Y-axis (G19)
E78: displacement in Y-axis (G17) or Z-axis (G18 or G19)

Remarks:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. With all distances the sign should be considered.
+: in positive direction of the axis.
-: in negative direction of the axis.
3. The feed is:
with G94 active in mm/min (or inches/min)
with G95 active in mm/rev (or inches/rev).
4. All parameters must be defined at the macro call.

Local parameters: E90 for the unconditional jump
E91 used with the relational expression
E92 to E99 different meanings

Tool sequence

1. Move the tool in the toolaxis to the position the clearance distance above the part surface. The spindle is stopped in the oriented position.
2. Move the tool out of the centre.
3. Move the tool to the point where the feed movement starts.
4. Move the tool back to the centre.
5. Start the spindle and execute the feed movement.
6. Stop the spindle in the oriented position after the dwell.
7. Move the tool out of the centre.
8. Retract the tool out of the hole; position defined by the retract distance
9. Move the tool back to the centre and start the spindle.

The macro

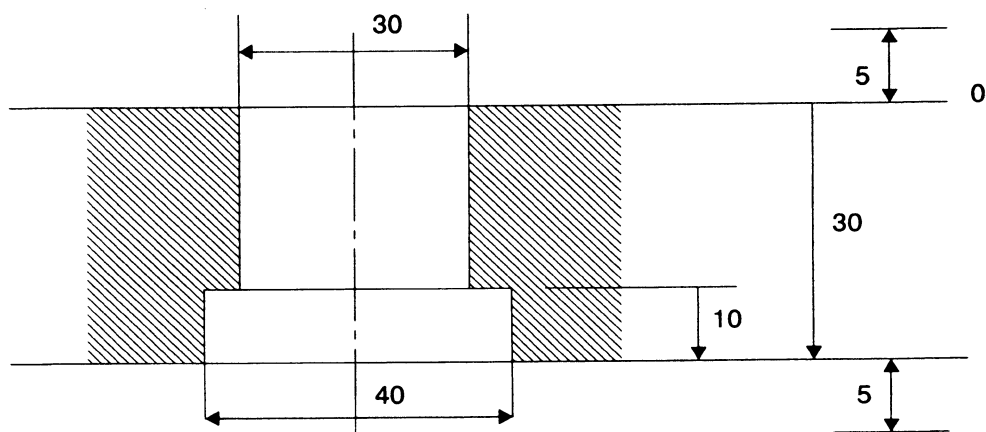
```

N99082      (BACK BORING CYCLE)
N1  E90=1 E95=E52 E97=E50 E98=E51 E99=29
N2  G29 E91=E60=17 E91 N=6
N3  E95=E51 E98=E52 E99=19
N4  G29 E91=E60=18 E91 N=6
N5  E95=E50 E97=E51 E99=9
N6  E92=E95+E64 E93=E92-E62 E92=E92-E70 E94=E95+E71
N7  E96=E95+E70 E97=E97+E77 E98=E98+E78
N8  G29 E90 K0 N=E99
N9  G0 X=E96 M19      (CYCLE IN X-AXIS)
N10 Y=E97 Z=E98
N11 X=E92
N12 Y=E51 Z=E52
N13 G1 X=E93 F=E73 S=E75 M=E76
N14 G4 X=E72 M19
N15 G0 Y=E97 Z=E98
N16 X=E94
N17 Y=E51 Z=E52
N18 G29 E90 K0 N=38
N19 G0 Y=E96 M19      (CYCLE IN Y-AXIS)
N20 X=E97 Z=E98
N21 Y=E92
N22 X=E50 Z=E52
N23 G1 Y=E93 F=E73 S=E75 M=E76
N24 G4 X=E72 M19
N25 G0 X=E97 Z=E98
N26 Y=E94
N27 X=E97 Z=E98
N28 G29 E90 K0 N=38
N29 G0 Z=E96 M19      (CYCLE IN Z-AXIS)
N30 X=E97 Y=E98
N31 Z=E92
N32 X=E50 Y=E51
N33 G1 Z=E93 F=E73 S=E75 M=E76
N34 G4 X=E72 M19
N35 G0 X=E97 Y=E98
N36 Z=E94
N37 X=E50 Y=E51
N38 S=E75 M=E76

```

Instruction for using the back boring cycle

1. The spindle position with the oriented stop must be known very well to the part programmer. He has to program a sign to the displacement values (E77 and E78), so the tool is properly positioned out of the hole.
If e.g. the tool stops at 12 o'clock, the displacement values can be E77=0 (X) and E78=-5 (Y). In this case the tool is positioned 5 mm out of the centre in the negative direction of the Y-axis.
2. With M19 the spindle is stopped in a well defined position. If an angular attachment is used for a tool in an other direction then the Z-axis, it depends on the machine tool, if the tool is also stopped in a well defined position.
3. At the moment of the call the part program must be in absolute dimension mode (G90-active).

Example of the use

NB8503

Fig. AN003.-2. Back boring on second wall

A back boring operation should be executed on the second wall of the hole from figure AN003.-2. Macro N99082 for executing this operation in the Z-axis is used.

It is assumed, that:

- the part surface is Z=0.
- the coordinates of the hole (XY-plane) are X0 and Y0.
- the spindle stops at 12 o'clock.

The partprogram could be:

```

N99182 (EXAMPLE OF BACK BORING OPERATION)
N1 G54 E60=17
N2 E50=0 E51=0 E52=15 T1 M6 (LOAD BORE)
N3 E61=0 E62=-10 E64=-30 E70=5 E72=0 E73=100
N4 E75=300 E76=3 E77=0 E78=-5 F100 S300 M3
N5 G22 N=99082
N6 M30
  
```

112.4 Linear pattern

Introduction

A special macro is designed for executing any machining cycle on equally spaced holes on a straight line.

It is possible to execute on this pattern:

- the fixed cycles of the Groundage control (G81 to G89),
- macros for machining cycles defined by Groundage,
- macros defined by the user.

Input parameters:

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

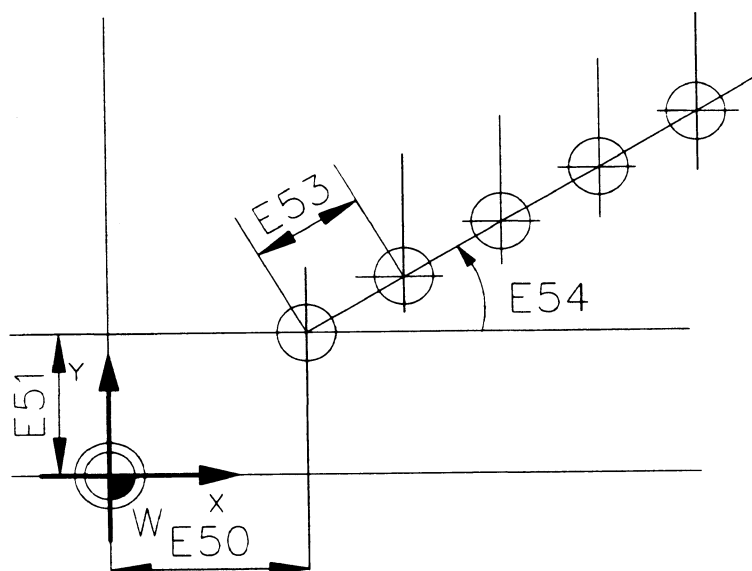


Fig. AN004.-1 Meaning of the input parameters

- E50: X-coordinate of first point (absolute)
 E51: Y-coordinate of first point (absolute)
 E52: Z-coordinate of first point (absolute)
- E53: distance between two points (without sign)
 E54: angle of the straight line with
 - the X-axis (G17 or G18)
 - the -Z-axis (G19)
 E55: number of holes (at least two) on the line.
- E59: = 0 the last defined fixed cycle of the control
 # 0 the identification number of the cycle macro
- E61: = 0 the machining cycle has to be executed on the first point of the pattern
 = 1 the first point is a reference point for the pattern; no machining cycle on that point

Remark:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. An angle is in degrees and decimal parts of a degree.
3. All parameters have to be defined at the macro call.

Output parameters:

- E50: X-coordinate of the cycle position
 E51: Y-coordinate of the cycle position
 E52: Z-coordinate of the cycle position
- E81: angle with the main axis (for pocket cycles)
- E110: the current number of the pattern position

Local parameters:

- E90: reserved for the unconditional jump
 E91: used with the relational expression
- E86 - E88: temporarily storage of E50 to E52
- E92 - E99: different meanings in the macro cycles
- E100 - E105: different meanings within the pattern macro

The macro

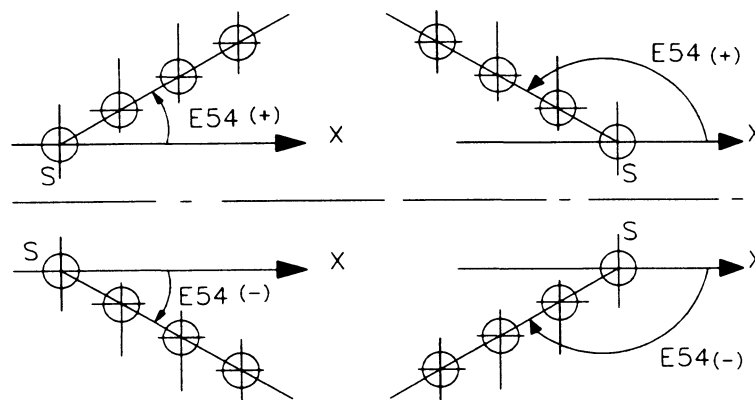
```

N99580 (LINEAR PATTERN)
N1 G29 E91=E101>3 E91 N=6
N2 E66=0 E81=E54 E90=1 E100=E55 1
N3 G29 E91=E101>1 E91 N=5
N4 E86=E50 E87=E51 E88=E52 E110=0
N5 E103=E53*cos(E54) E104=E53*sin(E54) E105=0
N6 G29 E91=E60=17 E91 N=11
N7 G29 E91=E60=18 E91 N=10
N8 E105=E103 E103=0
N9 G29 E90 K0 N=11
N10 E105=E104 E104=0
N11 G29 E91=E101>0 E91 N=13
N12 E50=E50-E103 E51=E51-E104 E52=E52-E105
N13 E50=E50+E103 E51=E51+E104 E52=E52+E105 E110=E110+1
N14 G29 E91=E61=E110 E91 N=19
N15 G29 E91=E59>0 E91 N=18
N16 G79 X=E50 Y=E51 Z=E52 B1=E66^E81
N17 G29 E90 K0 N=19
N18 G22 N=E59
N19 G29 E100 N=13
N20 G29 E91=E101=-1 E91 N=24
N21 G29 E91=E101<3 E91 N=23
N22 G29 E91=E101<>4 E91 N=24
N23 E50=E86 E51=E87 E52=E88
N24

```

Instruction for use

1. The pattern can have any position in the plane.

**Fig. AN004.-2 The location of the pattern**

The angle (E54) must be between -360° and 360°.
 The sign of the angle can be seen in the figure.
 Use always a positive sign for the distance E53.

2. The parameter E101 is used by other patterns (circular, grid, staggered grid or rhomboidal pattern) to jump over the check for storing the coordinates of the first point of the linear pattern. Parameter E101 should be set equal to 0 (E101=0) at the call of macro N99580.
3. Macros for machining cycles defined by the user can also be executed on the pattern assuming that the user macro has the same structure as a cycle macro defined by Groundage. The structure is:
 - calculate the position in the tool axis
 - move tool to the position where the cycle has to be executed. The positioning logic of the control (active in G0) can be used to avoid a collision between tool and workpiece.
 - execute the movements
 - . in the toolaxis for hole operations
 - . in the main plane for milling operations
4. A few macro numbers are used by Groundage to define some macros for machining cycles, e.g. N99082, N99084, N99085, N99086.
5. All fixed cycles of the Groundage control for the hole operations (G81, G83, G84, G85 and G86) and the cycle for milling a circular pocket (G89) can be executed on the pattern without any limitation. The use of the fixed cycles for milling a rectangular pocket (G87) or groove (G88) is restricted to a pocket or groove parallel to the main axes.

Example of the use

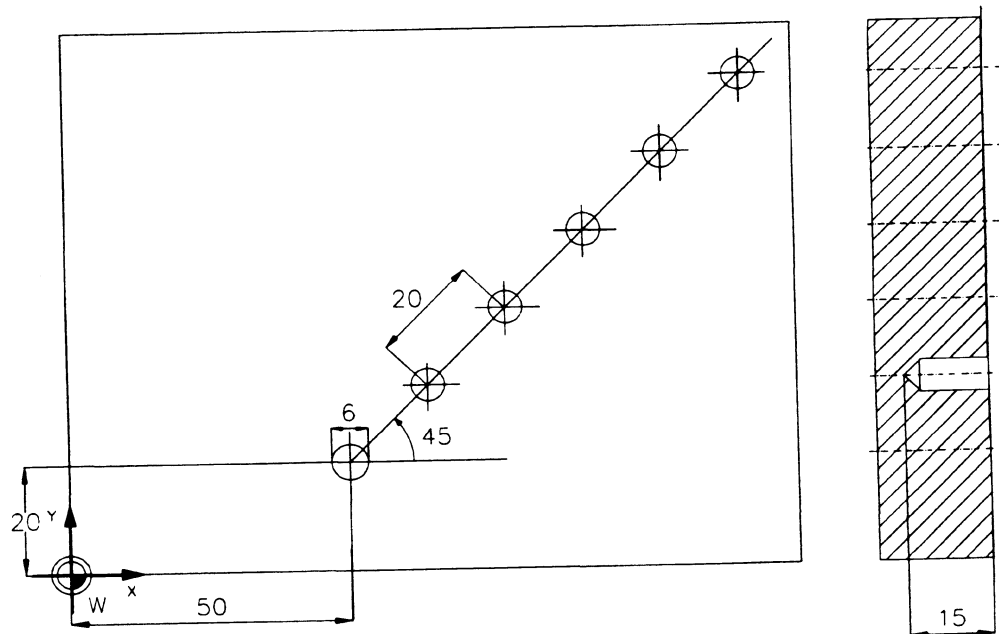


Fig. AN004.-3 A linear hole pattern

The holes of the pattern of figure AN004.-3 should be drilled and reamed. The fixed cycle (G81) of the control is used for drilling the holes and macro cycle N99085 for reaming. The part surface is the plane Z=0, defined by E52.

The partprogram could be:

N20980 (EXAMPLE LINEAR PATTERN)

```

N1  G54
N2  G98 X50 Y0 Z0 I150 J150 K20
N3  G99 X40 Y-10 Z-5 I170 J140 K-40
N4  T31 M6 E60=17
N5  G81 Y2 Z-15 F500 S1000 M3 E59=0
N6  E50=50 E51=20 E52=0 E53=20 E54=45 E55=6 E61=0
N7  G22 N=99580 E101=0
N8  T32 M6
N9  E59=99085 E62=-12 E70=2 E71=0 E72=0 E73=100 E74=500
N10 G22 N=99580 E101=0
N11 Z100 M30

```

Explanation

- N2/3: Define window (G98) and material for the graphical display of the pattern on the control.
- N4: Load the drill.
- N5: The fixed cycle (G81) of the Groundage control for drilling the holes is defined.
- N6: The parameters for defining the pattern are programmed
- N7: The macro for executing the drilling cycle on the linear pattern is called.
- N8: Load the reamer.
- N9: Set the parameters for the reaming macro N99085.
Refer to chapter Application Notes, reaming cycle with separate outfeed.
- N10: Execute the reaming macro on the linear pattern. It is not necessary to repeat the parameters which define the pattern.

112.5 Circular pattern

Introduction

A special macro is designed for executing any machining cycle on equally spaced holes on a circle.

It is possible to execute on this pattern:

- the fixed cycles of the Groundage control (G81 to G89),
- macros for machining cycles defined by Groundage,
- macros defined by the user.

Input parameters:

E60: Code for the plane

=17 for XY-plane

=18 for XZ-plane

=19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

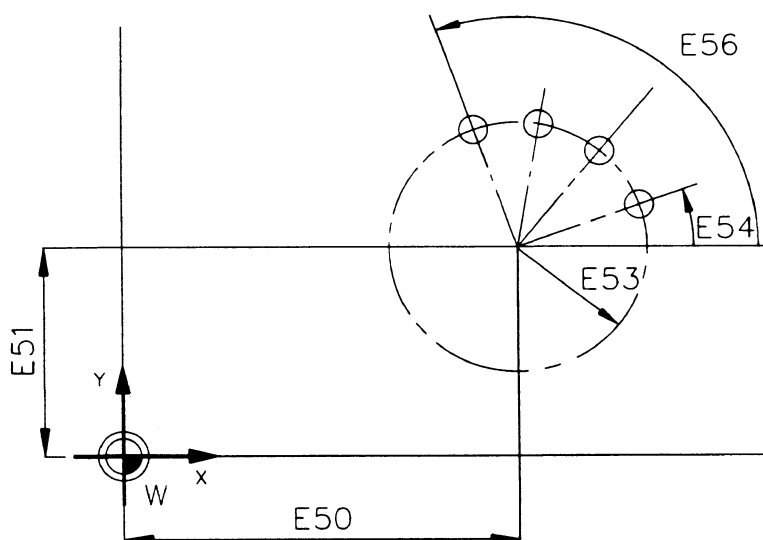


Fig. AN005.-1 Meaning of the input parameters

E50:	X-coordinate of the centre of the circle (absolute)
E51:	Y-coordinate of the centre of the circle (absolute)
E52:	Z-coordinate of the centre of the circle (absolute)
E53:	Radius of the circle
E54:	Starting angle
E55:	Number of holes on the are
E56:	Ending angle
E59:	= 0 the last defined fixed cycle of the control # 0 the identification number of the cycle macro
E61:	= 0 the machining cycle has to be executed on the first point of the pattern = 1 the first point is a reference point for the pattern; no machining cycle on that point
E66:	=0 with the fixed cycles (G81, G83, 684, G85, G86, G89) of the Groundage control a pocket or groove defined with the fixed cycle G87 or G88 resp. has to be milled parallel to the main axes. =1 a pocket (G87) or groove (G88) has to be milled in radial direction

Remark:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. An angle is in degrees and decimal parts of a degree.
3. All parameters have to be defined at the macro call.

Output parameters:

E50:	X-coordinate of the cycle position
E51:	Y-coordinate of the cycle position
E52:	Z-coordinate of the cycle position
E81:	angle with the main axis (for pocket cycles)
E110:	the current number of the pattern position

Local parameters:

E90:	reserved for the unconditional jump
E91:	used with the relational expression
E86- E88:	temporarily storage of E50 to E52
E92- E99:	different meanings in the macro cycles
E100 - E109:	different meanings within the pattern macro

Calling macro: N99580

Refer to Application Note AN004, dated 910220, for a description of macro N99580.

The macro**N99581 (CIRCULAR PATTERN)**

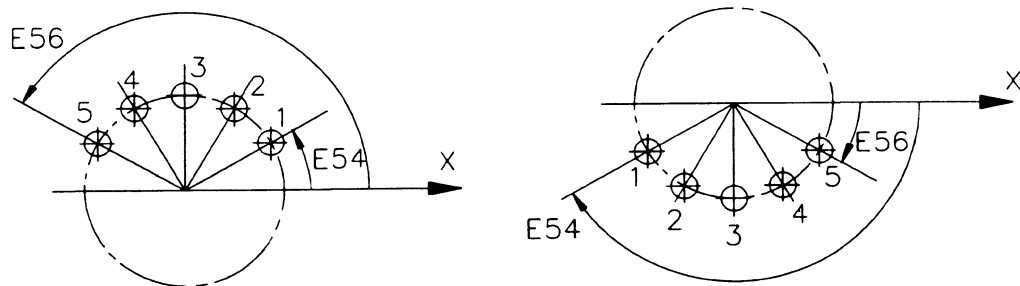
```

N1  E85=1 E102=E56-E54 E101=E55-1 E110=0
N2  G29 E91=abs(E102)<.1 E91 N=7
N3  G29 E91=(360-abs(E102))<.1 E91 N=5
N4  G29 E91=abs(E102-360)>.1 E91 N=8
N5  G29 E91=E102>0 E91 N=7
N6  E85=-1
N7  E102=E85*360 E101=E55
NS  E102=E102: E101
N9  E81=E54-E102 E85=E55-1 E106=cos(E102) E107=sin(E102)
N10 E86=E50 E87=E51 E88=E52 E90=1
N11 E103=E53*cos(E54) E104=E53*sin(E54)
N12 E81=E81+E102 E100=0 E105=0 E108=E103 E109=E104
N13 G22 N=99580 E101=4
N14 E103=E108*E106-E109*E107 E104=E108* E107+E109* E106
N15 G14 N1=12 N2=14 E85

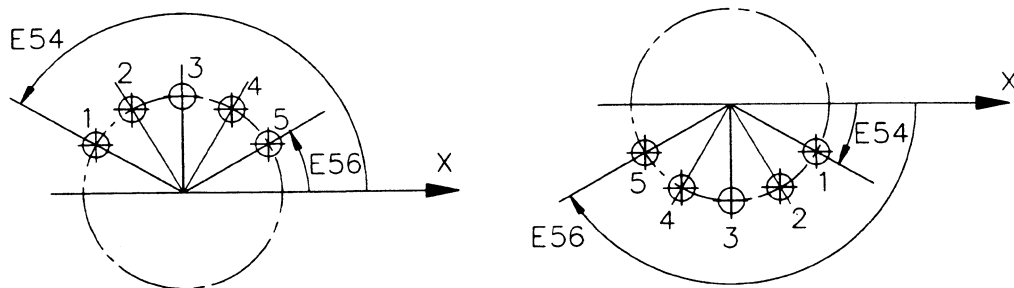
```

Instruction for use

- For processing holes on a circular arc
 - in counter clockwise direction just make the starting angle less than the ending angle

**Fig. AN005.-2 Holes processed in CCW direction**

- in clockwise direction make the starting angle greater than the ending angle

**Fig. AN005.-3 Holes processed in CW direction**

- For processing holes equally placed on a full circle
 - in counter clockwise direction just add 360° to the starting angle to find the ending angle or

- program the same value for both angles.
- in clockwise direction subtract 360° from the starting angle to find the ending angle

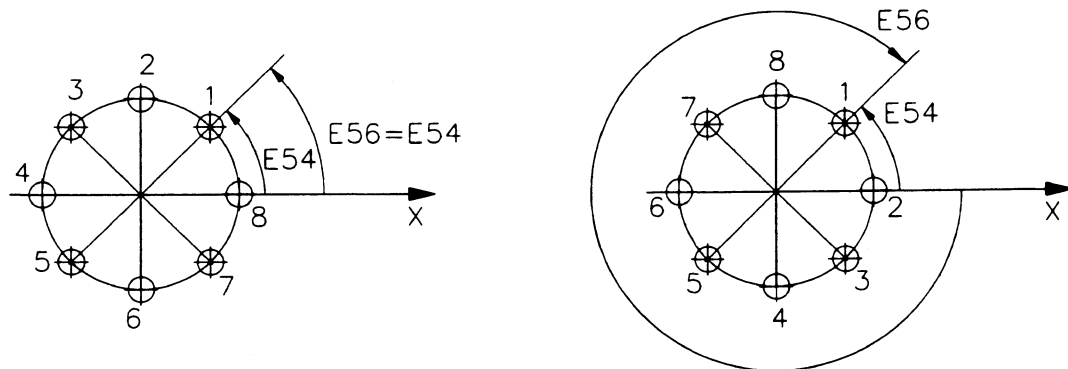


Fig. AN005.-4 Holes on a full circle

- Macros defined by the user can also be executed on the pattern assuming that the user macro has the same structure as a cycle macro defined by Groundage. The structure is:
 - calculate the position in the tool axis
 - move tool to the position where the cycle has to be executed. The positioning logic of the control (active in G0) can be used to avoid a collision between tool and workpiece.
 - execute the movements
 - . in the toolaxis for hole operations
 - . in the main plane for milling operations
- A few macro numbers are used by Groundage to define some macros for machining cycles, e.g. 99082, N99084, N99085, N99086.
- All fixed cycles of the Groundage control for the hole operations (G81, G83, G84, G85 and G86) and the cycle for milling a circular pocket (G89) can be executed on the pattern without any limitation. The use of the fixed cycles for milling a rectangular pocket (G87) or groove (G88) is restricted to a pocket or groove parallel to the main axes or in radial direction.
- Whilst using any fixed cycle of the Groundage control, parameter E66 should be defined as indicated with the input parameters. Parameter E66 can be ignored if it is not used as an input parameter of a macro cycle.

Example of the use

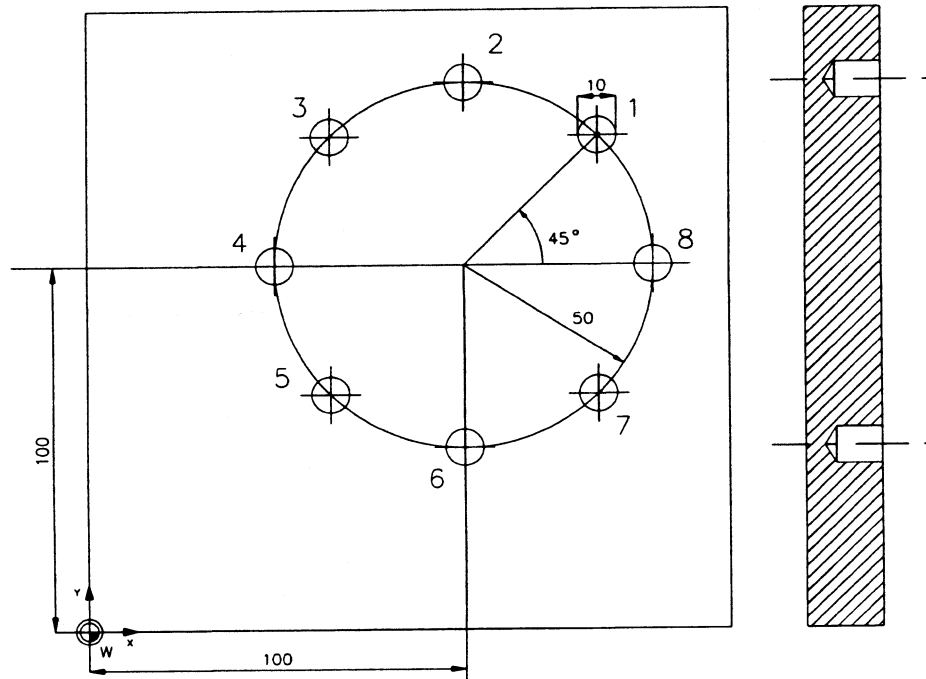


Fig. AN005.-5 A circular pattern

The holes of the pattern of figure AN005.-5 should be drilled and reamed. The fixed cycle (G81) of the control is used for drilling and macro cycle N99085 for reaming. The partsurface is the plane Z=0, defined by E52.

The program could be:

```

N20981      (EXAMPLE CIRCULAR PATTERN)
N1  G54
N2  G98 X0 Y0 Z0 1200 J200 K50
N3  G99 X-10 Y-10 Z-20 I220 J200 K50
N4  T31 M6 E60=17
N5  G81 Y2 Z-15 F500 S1000 M3 E59=0
N6  E50=100 E51=100 E52=0 E53=50 E54=45 E55=8
N7  E56=45 E61=0 E66=0
N8  G22 N=99581
N9  T32 M6
N10 E59=99085 E62=-12 E70=2 E71=0 E72=0 E73=100 E74=500 N11 G22 N=99581
N12 Z100 M30
  
```

Explanation

N2/3: Define window (G98) and material for the graphical display of the pattern on the control.

N4: Load the drill.

N5: The fixed cycle (G81) of the Groundage control for drilling the holes is defined.

N6/7: The parameters for defining the pattern are programmed

N8: The macro for executing the drilling cycle on the circular pattern is called.

N9: Load the reamer.

N10: Set the parameters for the reaming macro N99085.

112.6 Grid pattern

Introduction

A special macro is designed for executing any machining cycle on a grid pattern.

It is possible to execute on this pattern:

- the fixed cycles of the Groundage control (G81 - G89),
- macros for machining cycles defined by Groundage,
- macros defined by the user.

Input parameters:

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

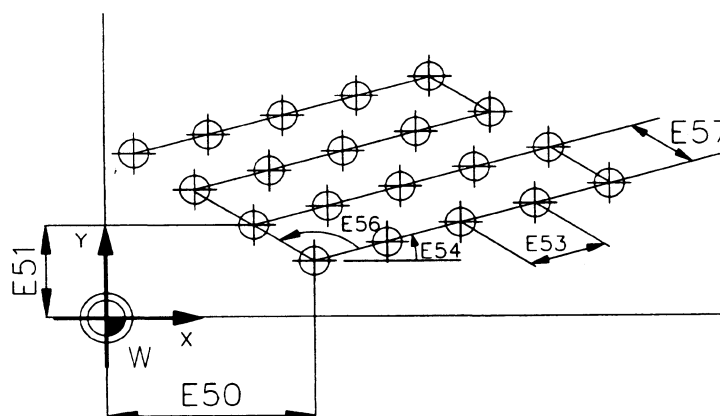


Fig. AN006.-1 Meaning of the input parameters

Application notes

- E50: X-coordinate of first point (absolute)
E51: Y-coordinate of first point (absolute)
E52: Z-coordinate of first point (absolute)
- E53: distance between two points on 1. line (no sign)
E54: angle of the first line with:
- the X-axis(G17 or G18)
- the -Z-axis (G19)
- E55: number of holes (at least two) on the first line
E56: angle between the two lines
E57: distance between two points on 2. line (no sign)
E58: number of holes (at least two) on the second line
- E59: = 0 the last defined fixed cycle of the control
0 the identification number of the cycle macro
- E61: = 0 the machining cycle has to be executed on the first point of the pattern
= 1 the first point is a reference point for the pattern; no machining cycle on that point

Remark:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. An angle is in degrees and decimal parts of a degree.
3. All parameters have to be defined at the macro call.

Output parameters:

- E50: X-coordinate of the cycle position
E51: Y-coordinate of the cycle position
E52: Z-coordinate of the cycle position
- E81: angle with the main axis (for pocket cycles)
- E110: the current number of the pattern position

Local parameters:

- E90: reserved for the unconditional jump
E91: used with the relational expression
- E86- E88: temporarily storage of E50 to E52
- E92- E99: different meanings in the macro cycles
- E100 - E109: different meanings within the pattern macro

Calling macro: N99580

Refer to chapter Application Notes, linear pattern.

The macro

N99582 (GRID PATTERN)

N1 E83=E53 E84=E54 E85=E55 E106=E58-2 E107=E54+E56

N2 G22 N=99580 E101=-1 (first line)

N3 E55=1 E53=E57 E54=E107

N4 G22 N=99580 E101=3 (FIRST POINT LINE)

N5 E53=E83 E54=E84+180 E55=E85-1

N6 G29 E91=E54<360 E91 N=8

N7 E54=E54-360

NS 622 N=99580 E84=E54 (NEXT LINE)

N9 G14 E106 N1=3 N2=8

N10 E50=E86 E51=E87 E52=E88 E54=E107-E56 E55=E85

Instruction for use

1. The pattern can have any position in the plane.

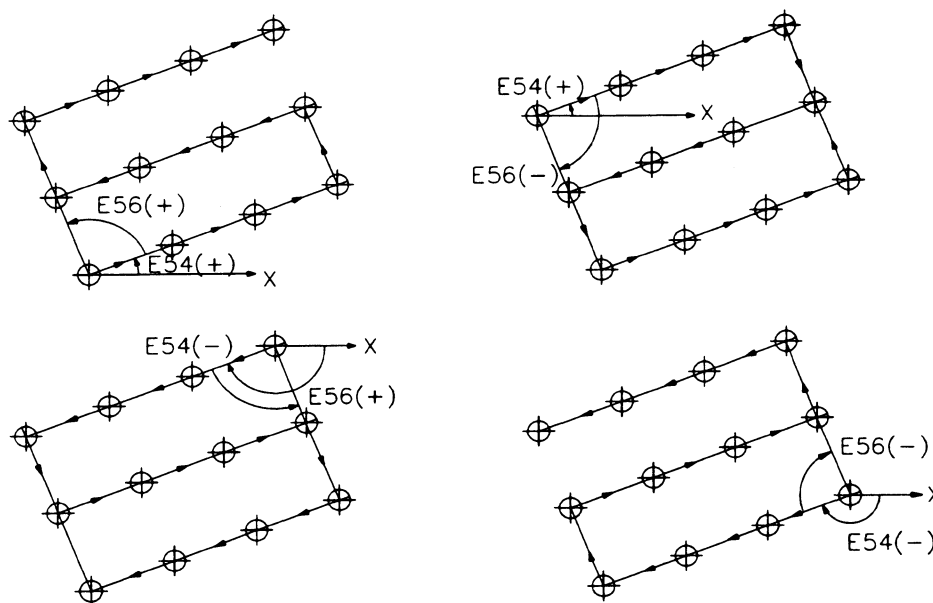


Fig. AN006.-2. Location of the pattern in the plane

The angles E54 and E56 must be between -360° and 360° . The sign of the angle can be seen in the figure.

Use always a positive sign for E53 and E57.

2. Macros defined by the user can also be executed on the pattern assuming that the user macro has the same structure as a cycle macro defined by Groundage. The structure is:
 - calculate the position in the tool axis
 - move tool to the position where the cycle has to be executed. The positioning logic of the control (active in G0) can be used to avoid a collision between tool and workpiece.
 - execute the movements
 - . in the toolaxis for hole operations
 - . in the main plane for milling operations
3. A few macro numbers are used by Groundage to define some macros for machining cycles, e.g. N99082, N99084, N99085, N99086.

4. All fixed cycles of the Groundage control for the hole operations (G81, G83, G84, G85 and G86) and the cycle for milling a circular pocket (G89) can be executed on the pattern without any limitation. The use of the fixed cycles for milling a rectangular pocket (G87) or groove (G88) is restricted to a pocket or groove parallel to the main axes.

Example of the use

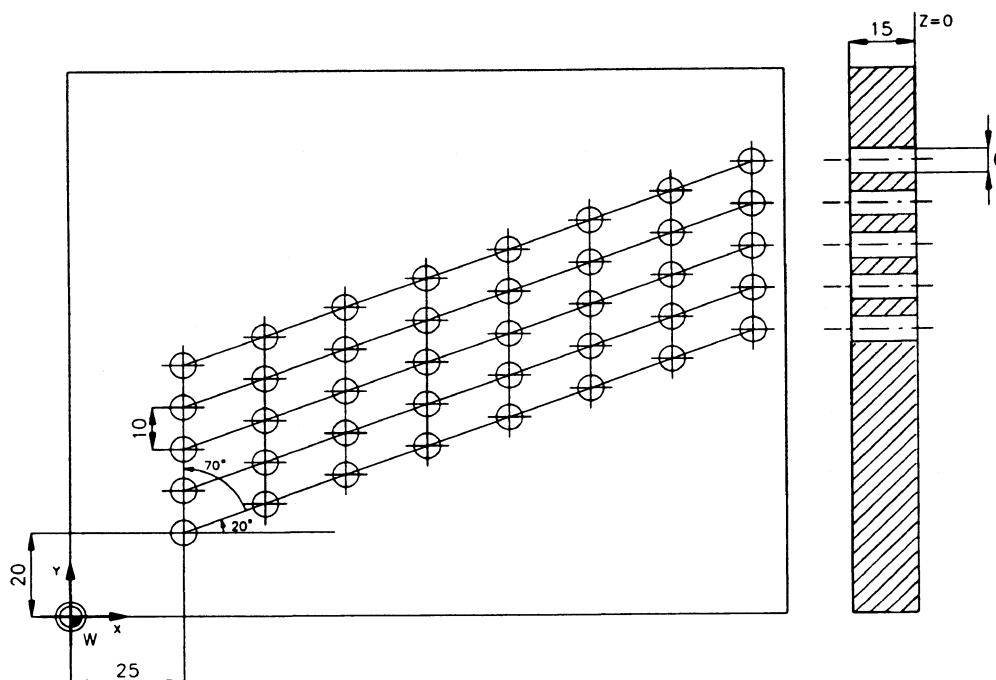


Fig. AN006.-3 A grid pattern

The holes of the pattern of figure AN006.-3 should be drilled and reamed. The fixed cycle (G81) of the control is used for drilling and macro cycle N99085 for reaming. The partsurface is the plane $Z=0$, defined by E52.

The partprogram could be:

N20982 (EXAMPLE GRID PATTERN)

```

N1 G54
N2 G98 X-20 Y-20 Z0 I200 J150 K50
N3 G99 X-15 Y-15 Z-20 I220 J150 K50
N4 T31 M6 E60=17
N5 G81 Y2 Z-15 F500 S1000 M3 E59=0
N6 E50=25 E51=20 E52=0 E53=20 E54=20 E55=8
N7 E56=70 E57=10 E58=6 E61=0
N8 G22 N=99582
N9 T32 M6
N10 E59=99085 E62=-12 E70=2 E71=0 E72=0 E73=100 E74=500
N11 G22 N=99582

```


Explanation

- N2/3: Define window (G98) and material for the graphical display of the pattern on the control.
- N4: Load the drill.
- N5: The fixed cycle (G81) of the Groundage control for drilling the holes is defined.
- N6/7: The parameters for defining the pattern are programmed
- N8: The macro for executing the drilling cycle on the grid pattern is called.
- N9: Load the reamer.
- N10: Set the parameters for the reaming cycle with separate outfeed macro N99085.
Refer to Groundage Application Note AN001, dated 900917 for a description of this macro and the meaning of the related parameters.
- N11: Execute the reaming macro on the grid pattern. It is not necessary to repeat the parameters which define the pattern.

112.7 Staggered grid pattern

Introduction

A special macro is designed for executing any machining cycle on a staggered grid pattern.

It is possible to execute on this pattern:

- the fixed cycles of the Groundage control (G81 to G89),
- macros for machining cycles defined by Groundage,
- macros defined by the user.

Input parameters:

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

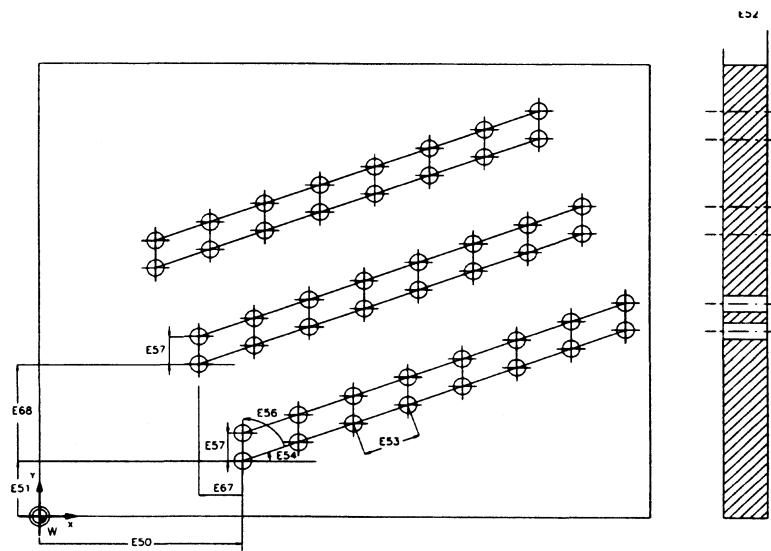


Fig. AN007-1 Meaning of the input parameters

E50:	X-coordinate of first point (absolute)
E51:	Y-coordinate of first point(absolute)
E52:	Z-coordinate of first point(absolute)
E53:	distance between two points on 1. line (no sign)
E54:	angle of the first line with: - the X-axis(G17 or G18) - the -Z-axis (G19)
E55:	number of holes on the parallel lines (at least two)
E56:	angle between the lines.
E57:	distance between the 1. and 2. line (no sign)
E58:	number of parallel lines (at least two)
E67:	grid shifting distance in X (G17/G18) or -Z (G19)
E68:	grid shifting distance in Y (G17/G19) or -Z (G18)
E59:	= 0 the last defined fixed cycle of the control # 0 the identification number of the cycle macro
E61:	= 0 the machining cycle has to be executed on the first point of the pattern = 1 the first point is a reference point for the pattern; no machining cycle on that point

Remark:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. An angle is in degrees and decimal parts of a degree.
3. All parameters have to be defined at the macro call.

Output parameters

E50:	X-coordinate of the cycle position
E51:	Y-coordinate of the cycle position
E52:	Z-coordinate of the cycle position
E81:	angle with the main axis (for pocket cycles)
E110:	the current number of the pattern position

Local parameters

E90:	reserved for the unconditional jump
E91:	used with the relational expression
E86- E88:	temporarily storage of E50 to E52
E92- E99:	different meanings in the macro cycles
E100 - E109:	different meanings within the pattern macro

Calling macro: N99580

Refer to chapter Application Notes, linear pattern.

THE MACRO

N99583 (STAGGERED GRID PATTERN)

```

N1  E83=E54+180 E84=E54 E85=E55 E106=E58-2 E107=E54+E56
N2  G29 E91=E83<360 E91 N=4
N3  E83=E83-360
N4  G22 N=99580 E101=-1 (FIRST STANDARD LINE)
N5  E100=0 E103=E67 E104=E68 E105=0
N6  G22 N=99580 E101=5 (FIRST POINT STAGGERED LINE)
N7  E54=E83 E55=E851
N8  G22 N=99580 E101=3 (STAGGERED LINE)
N9  G29 E91=E106<=0 E91 N=15
N10 E103=(-E57*cos(E107)+E67) E104=E57*sin(E107)+E68 E105=0
N11 G22 N=99580 E101=5 E100=0 (FIRST POINT STANDARD LINE)
N12 E54=E84 E55=E85-1 E106=E106-1
N13 G22 N=99580 E101=3 (NEXT STANDARD LINE)
N14 G29 E106 N=5
N15 E50=E86 E51=E87 E52=E88 E54=E84 E55=E85

```

Instruction for use

1. The pattern can have any position in the plane.

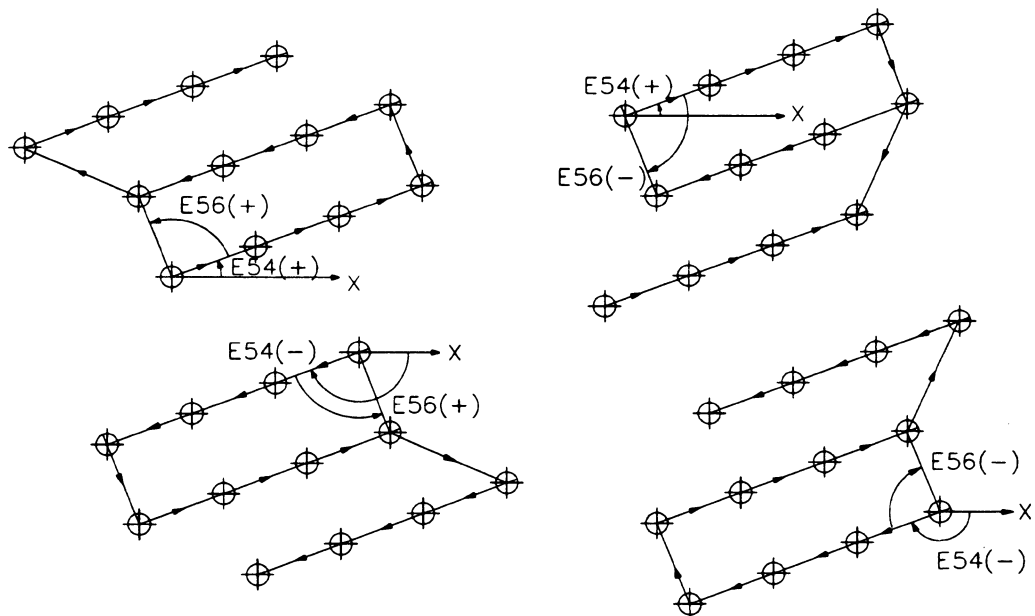


Fig. AN007-2. Location of the pattern in the plane

The angles E54 and E56 must be between -360° and 360° . The sign of the angle can be seen in the figure.

Use always a positive sign for E53 and E57.

2. Macros defined by the user can also be executed on the pattern assuming that the user macro has

the same structure as a cycle macro defined by Groundage. The structure is:

- calculate the position in the tool axis
 - move tool to the position where the cycle has to be executed. The positioning logic of the control (active in G0) can be used to avoid a collision between tool and workpiece.
 - execute the movements
 - . in the toolaxis for hole operations
 - . in the main plane for milling operations
3. A few macro numbers are used by Groundage to define some macros for machining cycles, e.g. N99092, N99084, N99085, N99086.
 4. All fixed cycles of the control for the hole operations (G81, G83, G84, G85 and G86) and the cycle for milling a circular pocket (G89) can be executed on the pattern without any limitation. The use of the fixed cycles for milling a rectangular pocket (G87) or groove (G88) is restricted to a pocket or groove parallel to the main axes.

Example of the use

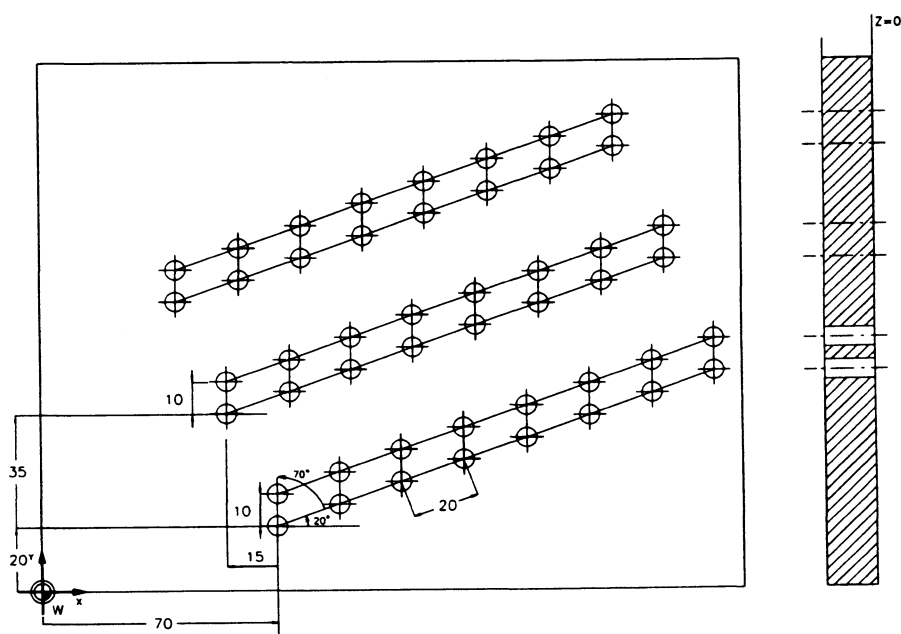


Fig. AN007-3 A staggered grid pattern

The holes of the pattern from figure AN007.-3 should be drilled and reamed. The fixed cycle (G81) of the control is used for drilling and macro N99085 for reaming.
The partsurface is the plane Z=0, defined by E52.

The partprogram could be:

N20983 (TEST STAGGERED GRID PATTERN)

```

N1  G54
N2  G98 X0 Y0 Z0 I250 J250 K50
N3  G99 X-10 Y-10 Z-20 I250 J250 K50
N4  Z50 T31 M6 E60=17
NS  G81 Y2 Z-15 F500 S1000 M3 E59=0
N6  E50=70 E51=20 E52=0 E53=20 E54=20 E55=8
N7  E56=70 E57=10 E58=6 E61=0 E67=-15 E68=35
N8  G22 N=99583
N9  Z50 T32 M6
N10 E59=99085 E62=-12 E70=2 E71=2 E72=0 E73=100 E74=500
N11 G22 N=99583
N12 Z100 M30
  
```

Explanation

N2/3:	Define the window (G98) and material for the graphical display of the pattern on the control.
N4:	Load the drill and define the toolaxis.
N5:	The fixed cycle (G81) of the Groundage control for drilling the holes is defined.
N6/N7:	The parameters for defining the pattern are programmed
N8:	The macro for executing the drilling cycle on the staggered grid pattern is called.
N9:	Load the reamer.
N10:	Set the parameters for the reaming macro N99085.
N11:	Execute the reaming macro on the staggered grid pattern. It is not necessary to repeat the parameters which define the pattern.

112.8 Finishing a circular pocket or island

Introduction

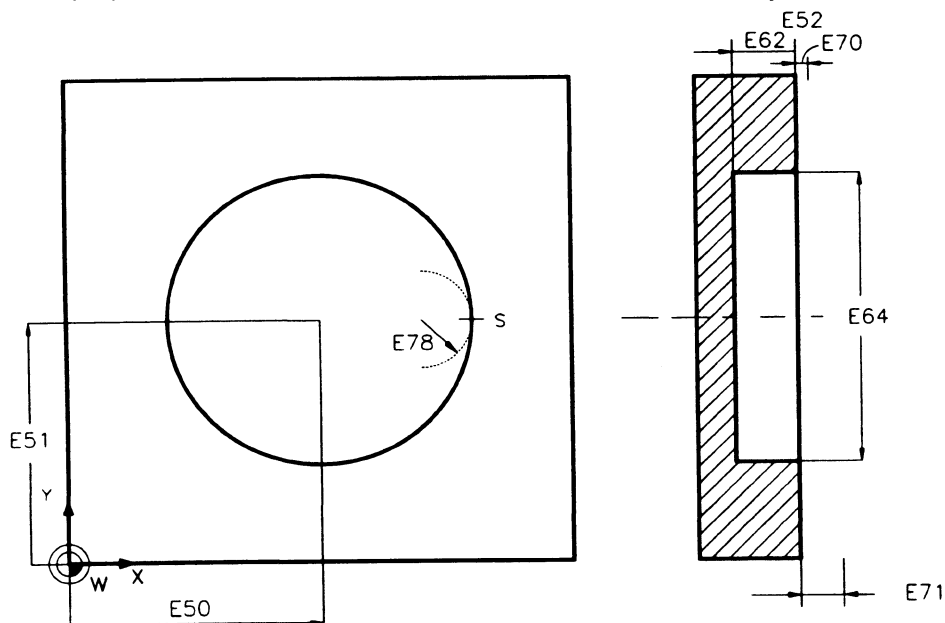
A special macro is designed for finishing a circular pocket or island.

Parameters used with this cycle

Input parameters

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.



AN008-1

Fig. AN008-1 Meaning of the input parameters (pocket)

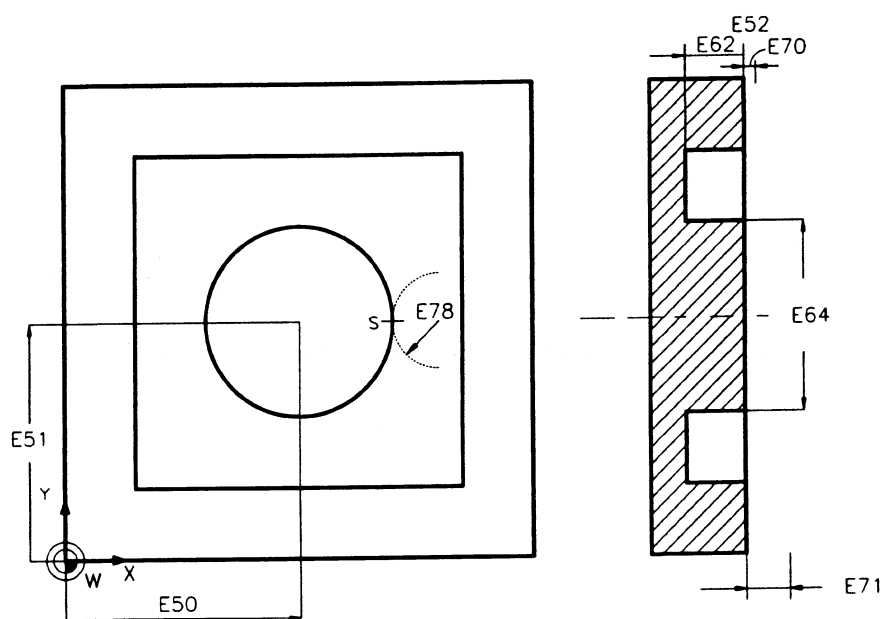


Fig. AN008-02 Meaning of the input parameters (island)

- E50: X-coordinate of the centre of the circle
 E51: Y-coordinate of the centre of the circle
 E52: Z-coordinate of the centre of the circle
- E63: type of part
 =0 a pocket
 =1 an island
- E62: depth of the pocket/ height of the island
 E64: diameter of the pocket or island
- E70: clearance distance
 E71: retract distance
- E73: feedrate for milling
 E74: indeed
- E75: direction of milling looking from the tool towards the workpiece
 =+1: conventional milling (counter clockwise)
 =-1: climb milling (clockwise)
- E78: radius of the entrance circle
 E48: tool diameter

Remarks:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches. Coordinates are always absolute.
2. With all distances the sign should be considered.
+: in positive direction of the axis.
-: in negative direction of the axis.
3. The feed is:
with G94 active in mm/min (or inches/min)
with G95 active in mm/rev (or inches/rev).
4. All parameters must be defined at the macro call.

Output parameters: None

Local parameters

E90: reserved for the unconditional jump
E91: used with the relational expression
E92- E99: different meanings
E111- E119: special meaning in the macros
E123: tool radius

Error detected

Plane -E60- not correctly defined

Tool diameter -E48- not defined

Calling macro (N99701): N99700 (Settings toolaxis for milling macros)

Called by: N99580, if any pattern is used.

N99702 (Milling a circular groove). Refer to chapter Application Notes, milling a groove macro.

The macros

SETTINGS TOOLAXIS FOR MILLING MACROS

N99700 E89=0 (SETTINGS TOOLAXIS FOR MILLING MACROS)

```
N1  E90=1 E95=E50 E96=E51 E97=E52 E111=(E75+5):2
N2  G29 E91=E60=17 E91 N=9
N3  E96=E52 E97=E51
N4  G29 E91=E60=18 E91 N=9
N5  G29 E91=E60=19 E91 N=8
N6  E89=1 M0 (PLANE -E60- NOT CORRECTLY DEFINED)
N7  G29 E90 K0 N=22
N8  E95=E51 E97=E50
N9  G=E60 E92=E97+E70 E93=E97+E62 E94=E97+E71 E95=E95+(E64:2)
N10 G29 E91=E115=0 E91 K0 N=16
N11 E77=abs(E77)
N12 G29 E91=abs(E62)>=E77 E91 N=14
N13 E77=abs(E62)
N14 E115=int(abs(E62):E77+.99) E116=E62:E115
N15 E115=E115-1
N16 G29 E91=E48>0 E91 N=19
N17 E89=1 M0 (TOOL DIAMETER -E48- IS NOT DEFINED)
N18 G29 E90 K0 N=22
N19 E123=E48-2
N20 G29 E91=E78>E123 E91 N=22
N21 E78=1.1*E123
N22
```

Finishing a circular pocket / island

N99701 (FINISHING A CIRCULAR POCKET/ISLAND)

```

N1  E99=15+(19-E60)*12 E117=0
N2  G29 E91=E63>=0 E91 N=7
N3  E63=abs(E63)-1 E99=E99+9*(1-E119)
N4  G29 E91=E63<2 E91 N=9
N5  E63=E63-2 E117=1
N6  G29 E90 K0 N=9
N7  G22 N=99700 E115=0
N8  G29 E89 K0 N=50
N9  G29 E91=E63>0 E91 N=12
N10 G29 E91=E78<(E64:2) E91 N=12
N11 E78=E64:4
N12 E112=E111-E63*E75 E113=(2*E111-5)*E78 E97=E96-E113
N13 E98=E96+E113 E113=44-E112 E114=E95+E78*(2*E63-1)
N14 G29 E90 K0 N=E99
N15 G0 X=E92 Y=E114 Z=E52
N16 G1 X=E93 F=E74
N17 G43 Z=E97 F=E73
N18 G=E113
N19 G=E112 Y=E95 Z=E52 J=E114 K=E52
N20 G=E111 J=E51 K=E52
N21 G=E112 Y=E114 Z=E98 J=E114 K=E52
N22 G40
N23 G29 E117 K0 N=50
N24 G0 Y=E114 Z=E52
N25 G0 X=E94
N26 G29 E90 K0 N=50
N27 G0 X=E114 Y=E92 Z=E52
N28 G1 Y=E93 F=E74
N29 G43 Z=E98 F=E73
N30 G=E113
N31 G=E112 X=E95 Z=E52 I=E114 K=E52
N32 G=E111 I=E50 K=E52
N33 G=E112 X=E114 Z=E97 I=E114 K=E52
N34 G40
N35 G29 E117 K0 N=50
N36 G0 X=E114 Z=E52
N37 G0 Y=E94
N38 G29 E90 K0 N=50
N39 G0 X=E114 Y=E51 Z=E92
N40 G1 Z=E93 F=E74
N41 G43 Y=E97 F=E73
N42 G=E113
N43 G=E112 X=E95 Y=E51 I=E114 J=E51
N44 G=E111 I=E50 J=E51
N45 G=E112 X=E114 Y=E98 I=E114 J=E51
N46 G40
N47 G29 E117 K0 N=50
N48 G0 X=E114 Y=E51
N49 G0 Z=E94
N50

```

Instruction for use

1. The starting point of the entrance circle is a fixed point (S) indicated in the figures AN008-01 and -02.
The part programmer has to look if he can use this point in his program.
2. If the programmed radius of the entrance circle (E78) is greater than the pocket radius, E78 is reduced in the macro to half the pocket radius.
3. If an error is detected in normal AUTO or SINGLE mode, a program stop (M₀) is executed and the error text displayed on the screen. After pressing START a jump to the end of the macro is executed and the macro is not processed any more.
Before the START the user can press MANUAL, so that after the START the control is in MANUAL mode. Now the user can cancel the execution of the program with MANUAL CLEAR CONTROL and change the program.
4. If an error is detected in a GRAPHICS mode of the control, the program stop M0 is executed, the macro number and the block number with the program stop is displayed. The error text itself is not displayed. So the user has to look in the listing of the macro to see which error is found. Leaving the graphics mode generates automatically a CLEAR CONTROL. Then the user can change the program and run it again.
5. An error detected in a macro, can not be found with MANUAL BLOCK SEARCH. If in this case a faulty parameter is used, the macro is not executed.
6. This macro can be used together with the Groundage macros for hole patterns. Refer to Application notes, boring cycle without dragline.

If an error is detected in any milling macro, the error is detected at each pattern point. All movements are ignored. Here too, program interrupt with MANUAL CLEAR CONTROL is recommended.
7. In macro N99700, line N9, the G-code for plane selection as defined with E60, is set. To avoid clearing the screen during any graphics mode, the proper G-code for plane selection (G17, G18 or G19) has to be programmed also before the functions G98 and G99.

Examples of the use

Example 1 Finishing a circular pocket

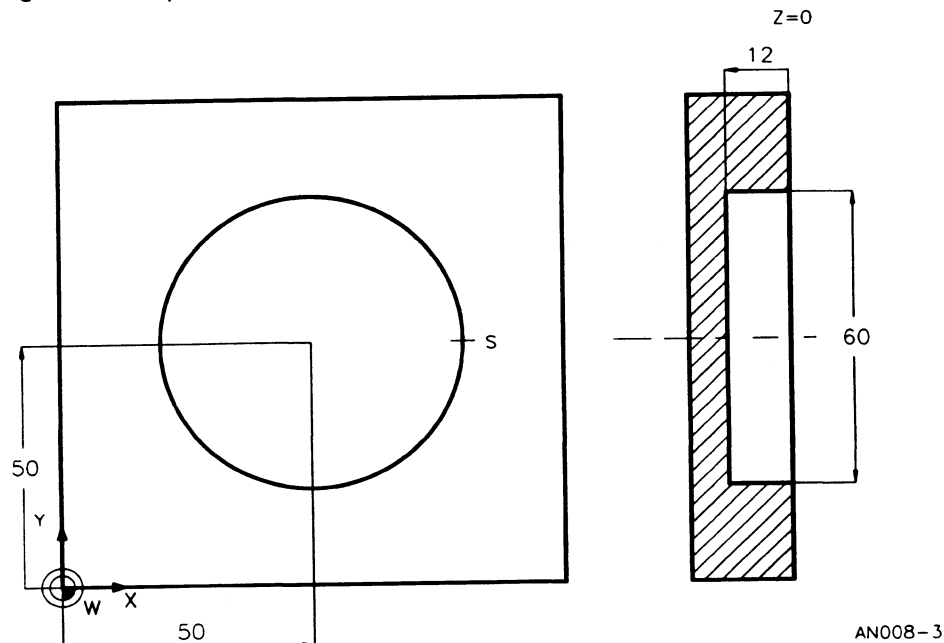


Fig. ANOOS-3 Example of finishing a pocket

Finishing the circular pocket from figure AN008-03 is requested. An endmill with a diameter of 14 mm is used.

The partprogram could be:

N207010 (FINISHING A CIRCULAR POCKET)

```

N1  G54
N2  G98 X0 Y0 Z0 I100 J100 K-30
N3  G99 X0 Y0 Z0 I100 J100 K-25
N4  Z10 T8 M6 E60=17 (ENDMILL DIAMETER 14)
N5  E50=50 E51=50 E52=0 E63=0 E62=-12 E64=60 E48=14
N6  E70=1 E71=1 E73=1000 E74=500 E75=1 E78=10
N7  G22 N=99701
N8  G0    Z100 M30
  
```

Explanation

- N2/3: Define the window (G98) and material dimensions for the graphical display of the pocket on the control.
- N4: Load the endmill and define the plane.
- N5: Enter the parameters for the location and the dimensions of the pocket.
- N6: Enter the parameters for finishing the pocket
- N7: Call the macro for finishing the pocket.
- N8: Retract the tool; end of the program.

Remark: If necessary, the G-code for plane selection is programmed between the lines N1 and N2 of the part program. Refer to point 7 of the Instruction for use.

Example 2 Finishing a circular island

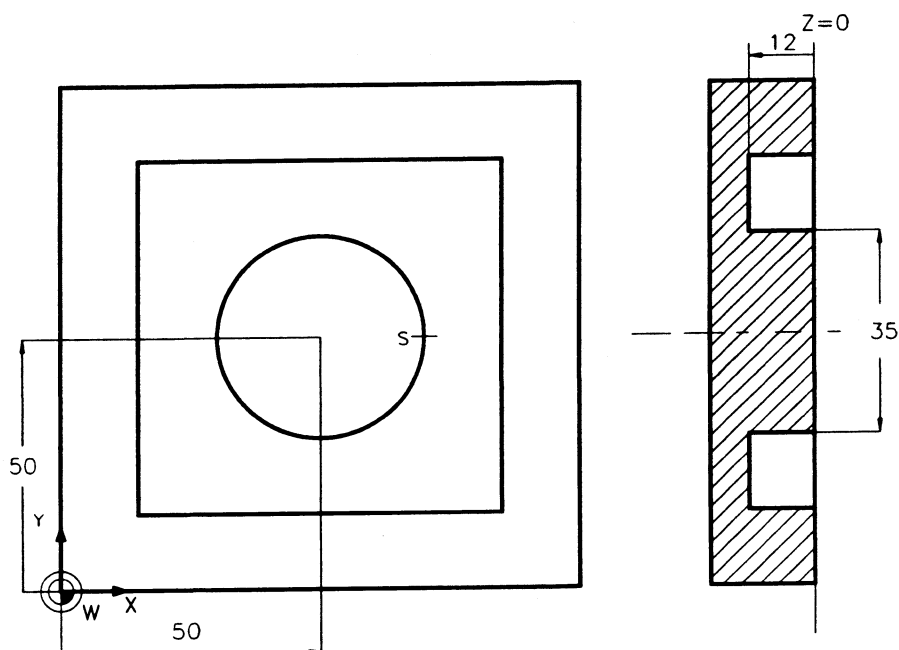


Fig. ANOOS-4 Example of finishing an island

Finishing the circular island from figure AN008-04 is requested. An endmill with a diameter of 14 mm is used.

The partprogram could be:

N207011 (FINISHING A CIRCULAR ISLAND)

```

N1  G54
N2  G98 X0 Y0 Z0 I100 J100 K-30
N3  G99 X0 Y0 Z0 I100 J100 K-25
N4  Z10 T8 M6 E60=17 (ENDMILL DIAMETER 14)
N5  E50=50 E51=50 E52=0 E63=1 E62= 12 E64=35 E48=14
N6  E70=1 E71=1 E73=1000 E74=500 E75=1 E78=10
N7  G22 N=99701
N8  G0    Z100 M30

```

Explanation

The only difference between the programs N207010 (POCKET) and N207011 is the value of E63 in line N5.

So refer to the description of that program.

112.9 Milling a circular groove

Introduction

A special macro is designed for milling a circular groove.

Parameters used with this cycle

Input parameters

E60: Code for the plane
=17 for XY-plane
=18 for XZ-plane
=19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

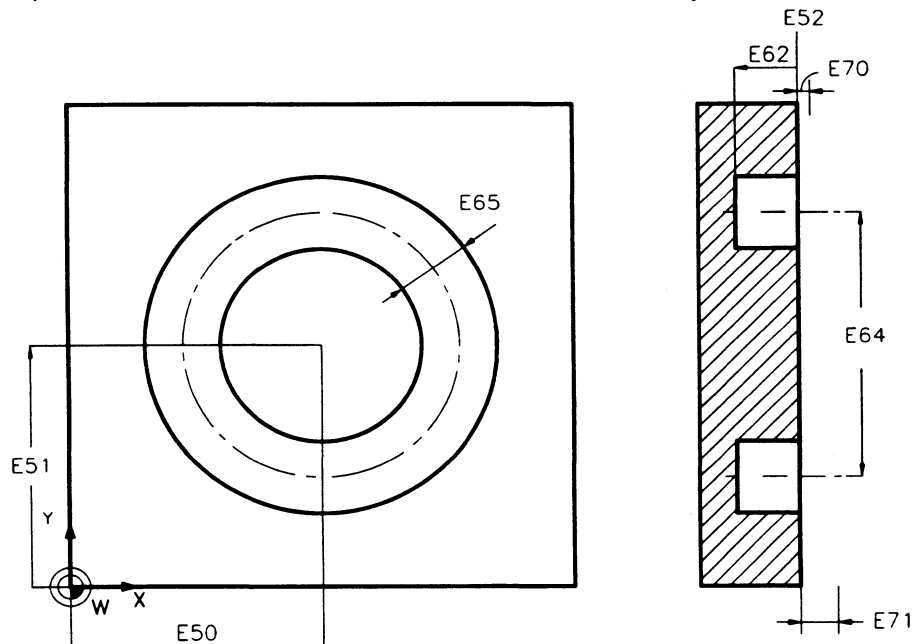


Fig. AN009-1 Meaning of the input parameters

Application notes

E50:	X-coordinate of the centre of the circle
E51:	Y-coordinate of the centre of the circle
E52:	Z-coordinate of the centre of the circle
E62:	depth of the groove
E64:	diameter of the pitch circle of the groove
E65:	width of the groove
E70:	clearance distance
E71:	retract distance
E73:	feedrate for milling
E74:	infeed
E75:	direction of milling looking from the tool towards the workpiece =+1: conventional milling(counter clockwise) =-1: climb milling(clockwise)
E77:	maximum step, if the milling takes place in different layers

Remarks:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches. Coordinates are always absolute.
2. With all distances the sign should be considered.
+: in positive direction of the axis.
-: in negative direction of the axis.
3. The feed is:
with G94 active in mm/min (or inches/min)
with G95 active in mm/rev (or inches/rev).
4. All parameters must be defined at the macro call.

Output parameters

E89: error indicator
=1: an error detected in the macro
=0: no error

Local parameters

E90: reserved for the unconditional jump
E91: used with the relational expression
E92 - E99: different meanings
E111 - E119: special meaning in this macro or the calling ones

Error detected

Plane (E60) not correctly defined

Calling macro: N99700 (Settings toolaxis for milling macros)
N99701 (Finishing pocket / island)

Called by: N99580, if any pattern is used.

The macro

N99702 (MILLING A CIRCULAR GROOVE)

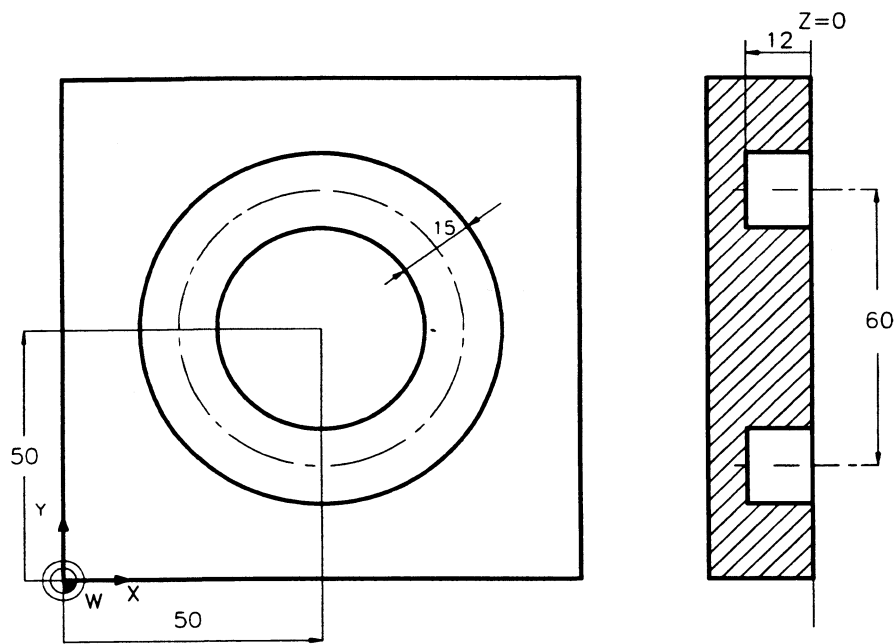
```

N1 G22 N=99700 E115=1 E78=E65:2
N2 G29 E89 K0 N=25
N3 E93=E97+E116 E99=5+(19-E60)*6
N4 G29 E90 K0 N=E99
N5 G0 X=E92 Y=E95 Z=E52
N6 G1 X=E93 F=E74
N7 G=E111 J=E51 K=E52 F=E73
N8 G1 E93=E93+E116
N9 G14 J=E115 N1=6 N2=8
N10 G29 E90 K0 N=22
N11 G0 X=E95 Y=E92 Z=E52
N12 G1 Y=E93 F=E74
N13 G=E111 I=E50 K=E52 F=E73
N14 G1 E93=E93+E116
N15 G14 J=E115 N1=12 N2=14
N16 G29 E90 K0 N=22
N17 G0 X=E95 Y=E51 Z=E92
N18 G1 Z=E93 F=E74
N19 G=E111 I=E50 J=E51 F=E73
N20 G1 E93=E93+E116
N21 G14 J=E115 N1=18 N2=20
N22 E93=E93-E116 E92=E93 E119=1
N23 G22 N=99701 E63=-3 E95=E95+E78
N24 G22 N=99701 E63=-2 E95=E95-E65
N25

```

Instruction for use

1. The starting point for milling the groove is indicated in figure AN009-01
2. After moving the tool to the first depth (see next point) the mill is moved along the pitch circle of the groove.
A number of paths is executed until the final depth is reached.
3. The step per depth is recalculated to an average value, so that each step is executed with the same value. In this way it is avoided that the final step is very small.
4. If the programmed stepsize is greater than the depth, the groove is milled in one step.
5. If the stepsize is negative, it is changed to a positive value
6. It is assumed that after milling the groove at depth, not too much material is left on the walls, so that they can be milled in one step. First the outer wall and thereafter the inner one is milled.
7. This macro can be used together with the Groundage macros for hole patterns. Refer to Groundage Application Bulletin 02.
8. Refer to chapter Application Notes, section finishing a circular pocket
9. Refer also to Application Note AN008, date 910620, for the instructions how to handle, if in macro N99700 an error is found.

Example of the use**Fig. AN009-2 Example of a circular groove**

The groove from figure AN009-02 should be milled. An endmill with a diameter of 14 mm is used.

The partprogram could be:

```

N20702    (EXAMPLE OF A CIRCULAR GROOVE)
N1  G54
N2  G98 X0 Y0 Z0 I100 J100 K-30
N3  G99 X0 Y0 Z0 I100 J100 K-25
N4  Z10 T8 M6 E60=17 (ENDMILL DIAMETER 14)
N5  E50=50 E51=50 E52=0 E62=-12 E64=60 E65=15
N6  E70=1 E71=1 E73=1000 E74=500 E75=1 E77=4
N7  G22 N=99702
N8  G0    Z100 M30
  
```

Explanation

- N2/3: Define the window (G98) and material dimensions for the graphical display of the groove on the control.
- N4: Load the endmill and define the plane.
- N5: Enter the parameters for the location and the dimensions of the groove.
- N6: Enter the parameters for milling the groove
- N7: Call the macro for milling the groove.
- N8: Retract the tool; end of the program.

112.10 Rhomboidal pattern

Introduction

A special macro is designed for executing any machining cycle on a rhomboidal pattern.

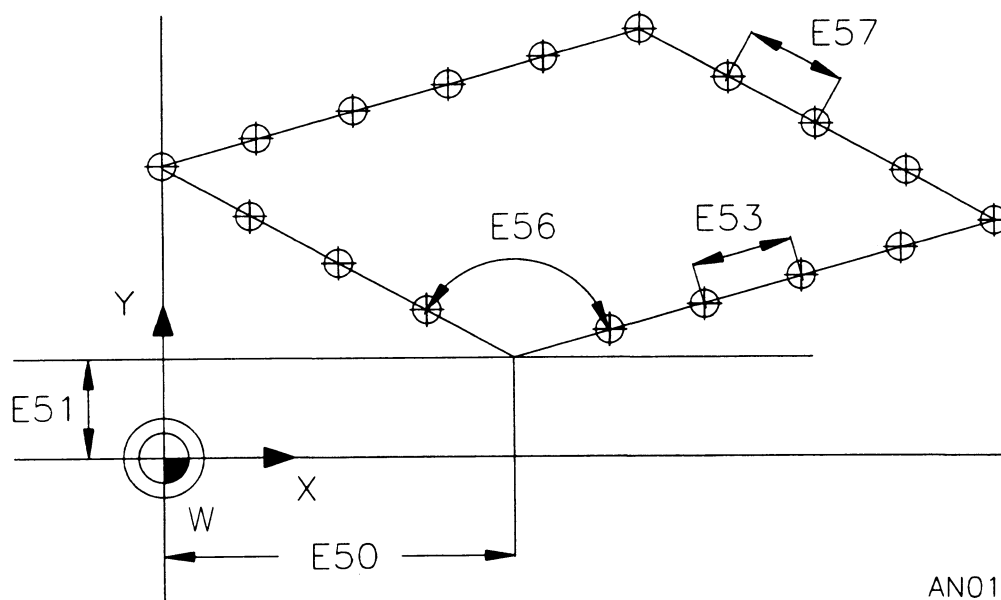
It is possible to execute on this pattern:

- the fixed cycles of the Groundage control (G81 to G89),
- macros for machining cycles defined by Groundage,
- macros defined by the user.

Input parameters:

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.



AN010-1

Fig. AN010-1 Meaning of the input parameters

- E50: X-coordinate of first point (absolute)
 E51: Y-coordinate of first point (absolute)
 E52: Z-coordinate of first point (absolute)
- E53: distance between two points on 1. line (no sign)
 E54: angle of the first line with:
 - the X-axis (G17 or G18)
 - the -Z-axis (G19)
- E55: number of holes (at least two) on the first line
 E56: angle between the two lines
 E57: distance between two points on 2. line (no sign)
 E58: number of holes (at least two) on the second line
- E59: = 0 the last defined fixed cycle of the control
 # 0 the identification number of the cycle macro
- E61: = 0 the machining cycle has to be executed on the first point of the pattern
 = 1 the first point is a reference point for the pattern; no machining cycle on that point

Remark:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches.
2. An angle is in degrees and decimal parts of a degree.
3. All parameters have to be defined at the macro call.

Output parameters

- E50: X-coordinate of the cycle position
 E51: Y-coordinate of the cycle position
 E52: Z-coordinate of the cycle position
- E81: angle with the main axis (for pocket cycles)
- E110: the current number of the pattern position

Local parameters

- E90: reserved for the unconditional jump
 E91: used with the relational expression
- E86 -E88: temporarily storage of E50 to E52
- E92 -E99: different meanings in the macro cycles
- E100 -E109: different meanings within the pattern macro

Calling macro: N99580
 Refer to chapter Application Notes section linear pattern.

The macro

N99584 (RHOMBOIDAL PATTERN)

N1 E83=E53 E84=E54 E85=E55 E107=E54+E56

N2 G22 N=99580 E101=-1 (FIRST LINE)

N3 E53=E57 E54=E107 E55=E58-1

N4 G22 N=99580 E101=3 (SECOND LINE)

N5 E53=-E83 E54=E84 E55=E85-1

N6 G22 N=99580 (THIRD LINE)

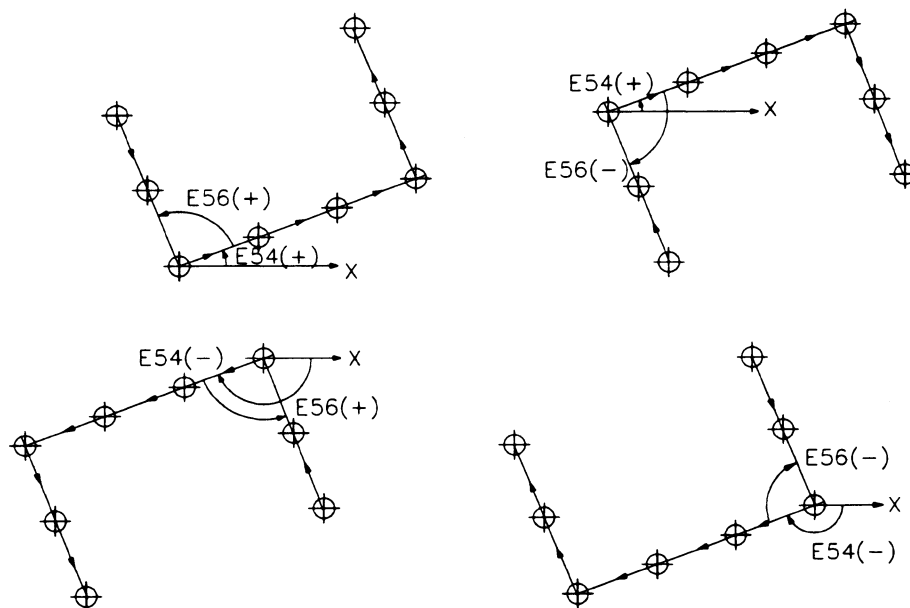
N7 E53=-E57 E54=E107 E55=E58-2

N8 G22 N=99580 E101=2 (FOURTH LINE)

N9 E53=E83 E54=E84 E55=E85

Instruction for use

1. The pattern can have any position in the plane.



AN010-2

Fig. AN010-2. Location of the pattern in the plane

The angles E54 and E56 must be between -360° and 360° . The sign of the angle can be seen in the figure.

Use always a positive sign for E53 and E57.

2. Macros defined by the user can also be executed on the pattern assuming that the user macro has the same structure as a cycle macro defined by Groundage. The structure is:
 - calculate the position in the tool axis
 - move tool to the position where the cycle has to be executed. The positioning logic of the control (active in G0) can be used to avoid a collision between tool and workpiece.
 - execute the movements
 - . in the toolaxis for hole operations
 - . in the main plane for milling operations
3. A few macro numbers are used by Groundage to define some macros for machining cycles, e.g. N99082, N99084, N99085, N99086.
4. All fixed cycles of the Groundage control for the hole operations (G81, G83, G84, G85 and G86) and the cycle for milling a circular pocket (G89) can be executed on the pattern without any limitation. The use of the fixed cycles for milling a rectangular pocket (G87) or groove (G88) is restricted to a pocket or groove parallel to the main axes.

Example of the use

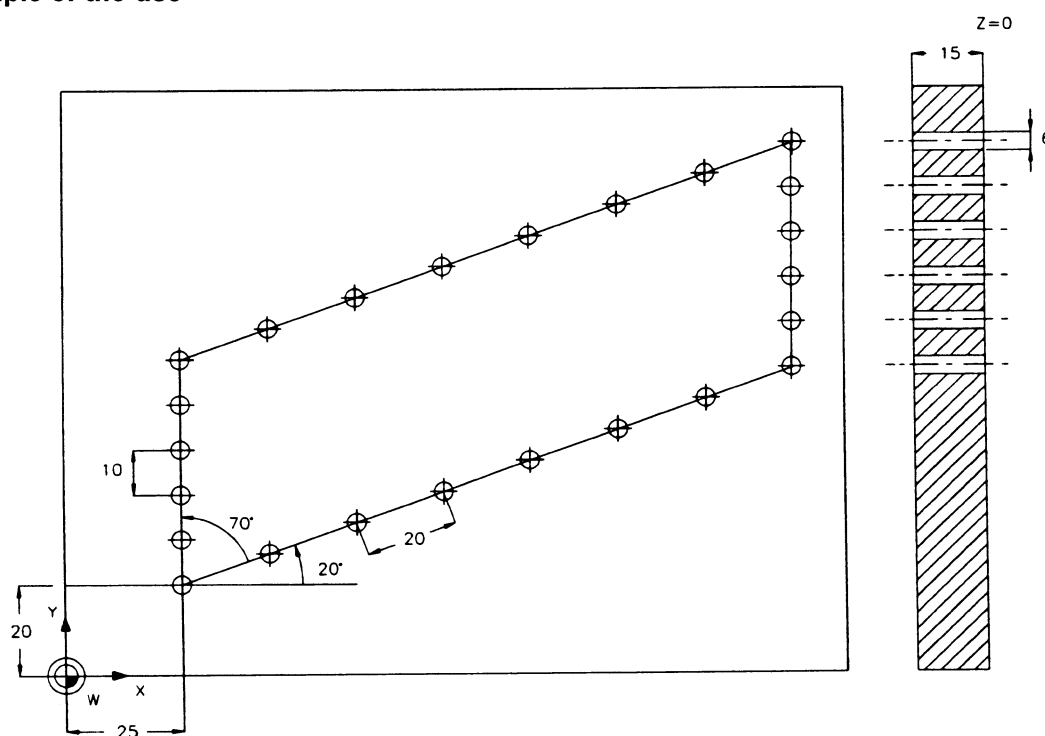


Fig. AN010-3 A rhomboidal pattern

AN010-3

The holes of the pattern of figure AN008-3 should be drilled and reamed. The fixed cycle (G81) of the control is used for drilling the holes and macro cycle N99085 for reaming. The part surface is the plane $Z=0$, defined by E52.

The partprogram could be:

N20984 (TEST RHOMBOIDAL PATTERN)

```

N1  G54
N2  G98 X0 Y0 Z0 I200 J150 K50
N3  G99 X-10 Y-10 Z-20 I220 J150 K50
N4  T31 M6 E60=17
N5  G81 Y2 Z-15 F500 S1000 M3 E59=0
N6  E50=25 E51=20 E52=0 E53=20 E54=20 E55=8
N7  E56=70 E57=10 E58=6 E61=0
N8  G22 N=99584
N9  T32 M6
N10 E59=99085 E62=-12 E70=2 E71=2 E72=0 E73=100 E74=500
N11 G22 N=99584
N12 Z100          M30

```

Explanation

N2/3: Define window (G98) and material for the graphical display of the pattern on the control.
 N4: Load the drill and define the tool axis.
 N5: The fixed cycle (G81) of the Groundage control for drilling the holes is defined.
 N6/7: The parameters for defining the pattern are programmed
 N8: The macro for executing the drilling cycle on the rhomboidal pattern is called.
 N9: Load the reamer.
 N10: Set the parameters for the reaming macro N99085.
 Refer to chapter Application Notes AN001, section rhomboidal pattern for a description of this macro and the meaning of the related parameters.
 N11: Execute the reaming macro on the rhomboidal pattern. It is not necessary to repeat the parameters which define the pattern.

112.11 Roughing a circular pocket

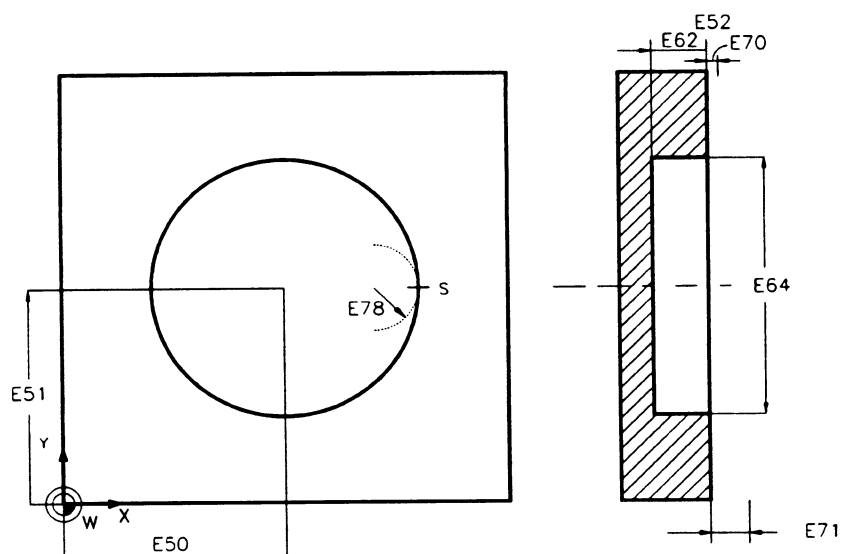
Introduction

A special macro is designed for roughing a circular pocket. The tool is moving in either a circle or a spiral. In the latter case better cutting conditions are achieved. The diameter of an existing hole (pre drilled or already available on the part) can be entered and is considered with the milling operation.

Input parameters

E60: Code for the plane
 =17 for XY-plane
 =18 for XZ-plane
 =19 for YZ-plane

Parameter E60 should be set once in the program. As long as the plane is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.



AN008-1

Figure AN011-1 Meaning of the input parameters

E50: X-coordinate of the centre of the circular pocket
E51: Y-coordinate of the centre of the circular pocket
E52: Z-coordinate of the centre of the circular pocket
E62: depth of the pocket
E64: diameter of the pocket

E69: diameter of
- an existing hole on the part
- a predrilled hole

=0, if a hole does not exist

E70: clearance distance
E71: retract distance

E72: way of milling
=0 standard circle milling and no finishing
=1 standard circle milling and finishing
=2 spiral milling and no finishing
=3 spiral milling and finishing

E73: feedrate for milling
E74: infeed
E75: direction of milling looking from the tool towards the workpiece
=+1: conventional milling (counter clockwise)
=-1: climb milling (clockwise)

E76: overlap parameter (in % of the tool diameter)

E77: maximum step for milling in different layers

E78: radius of the entrance circle
E79: finishing allowance

E48: diameter of the mill

Remarks:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches. Coordinates are always absolute.
2. With all distances the sign should be considered.
+: in positive direction of the axis.
-: in negative direction of the axis.
3. The feed is:
with G94 active in mm/min (or inches/min)
with G95 active in mm/rev (or inches/rev).
4. All parameters must be defined at the macro call.

Output parameters

E89: error indicator
=1: an error detected in the macro
=0: no error

Local parameters

E90: reserved for the unconditional jump
E91: used with the relational expression
E92 - E99: different meanings with the cycles
E100 - E110: special meanings in the pattern macros
E111 - E119: special meaning in macro N99701
E121 - E125: input parameters for the spiral macro N99550
E126 - E127: special meaning in macro N99703
E128 - E139: special meaning in macro N99550

Error detected

Plane - E60 - not correctly defined

Tool diameter - E48 - not defined

Diameter of the existing hole - E69 - greater than the pocket diameter

Calling macro

N99700 (Settings toolaxis for milling macros)
N99550 (Milling spiral of Archimedes)
N99701 (Finishing pocket / island)

Called by

N99580 General pattern macro, if any pattern is used

N99704 (Circular pocket with spherical bottom)

THE MACROS**MILLING A SPIRAL OF ARCHIMEDES**

```
N 99550 (MILLING A SPIRAL OF ARCHIMEDES)
N1 G29 E91=abs(E122-E121)<=abs(E124) E91 N=21
N2 G29 E91=E123<>0 E91 N=5
N3 E89=1 M0 (RADIUS INCREASE -E123- EQUALS 0)
N4 G29 E89 K0 N=21
N5 F=E73 E128=abs(E123) E129=abs(E124) E130=30-E60 E131=E121
N6 G29 E91=E121<E122 E91 N=8
N7 E128=-E128 E129=-E129
N8 E132=E75*E129*360:E128 E135=cos(E132) E136=sin(E132)
N9 E133=E121*cos(E121*E128:360) E134=E121*sin(E121*E128:360)
N10 G29 K0 E90 N=14
N11 G1 Y=E51+E134 Z=E52-E133
N12 G1 X=E50+E133 Z=E52-E134
N13 G1 X=E50+E133 Y=E51+E134
N14 G14 N1=E130
N15 E137=1+E129:E131 E138=E137*E133 E139=E137*E134 E131=E131+E129
N16 E133=E138*E135-E139*E136 E134=E139*E135+E138*E136
N17 G29 E91 N=14 E91=abs(E122-E131)>abs(E124)
N18 G29 E91=E125=0 E91 N=21
N19 E137=E122:E131 E133=E133*E137 E134=E134*E137
N20 G14 N1=E130
N21
```

ROUGHING A CIRCULAR POCKET

N99703 (ROUGHING A CIRCULAR POCKET)

```

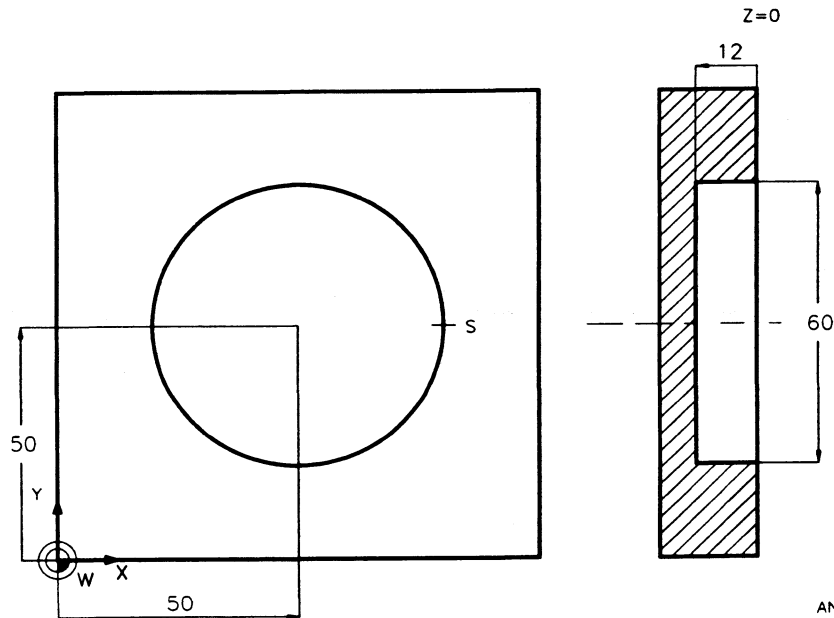
N1  G29 E91=E64<=0 E91 N=7
N2  G22 N=99700 E115=1
N3  G29 E89 K0 N=42
N4  G29 E91=E69<=E64 E91 N=7
N5  E89=1 M0 (DIAMETER OF THE EXISTING HOLE - E69- GREATER THAN THE POCKET
      DIAMETER)
N6  G29 E89 K0 N=42
N7  E99=12+(19-E60)*3 E119=E72 E127=E99+17 E64=abs(E64)
N8  E93=E97+E116 E122=.5*E64-E79-E123 E125=E69:2
N9  G29 E91=E69>=E48 E91 N=11
N10 E125=E48:2
N11 G29 E90 K0 N=E99
N12 G0 X=E92 Y=E51 Z=E52
N13 G1 X=E93 F=E74
N14 G29 E90 K0 N=20
N15 G0 X=E50 Y=E92 Z=E52
N16 G1 Y=E93 F=E74
N17 G29 E90 K0 N=20
N18 G0 X=E50 Y=E51 Z=E92
N19 G1 Z=E93 F=E74
N20 G29 E91=E72<2 E91 N=25
N21 E119=E72-2 E121=E125 E124=E123:60
N22 G29 E91=E121>=E122 E91 N=27
N23 G22 N=99550 E126=39 E127=E127+1
N24 G29 E90 K0 N=28
N25 E121=E125+.01*E76^E48-E123 E126=26
N26 G29 E91=E121<E122 E91 N=28
N27 E121=E122 E126=39
N28 G29 E90 K0 N=E127
N29 G1 Y=(E51+E121) F=E73
N30 G=E111 J=E51 K=E52
N31 G29 E90 K0 N=37
N32 G1 X=(E50+E121) F=E73
N33 G=E111 I=E50 K=E52
N34 G29 E90 K0 N=37
N35 G1 X=(E50+E121) F=E73
N36 G=E111 I=E50 J=E51
N37 E121=E121+.01*E76^E48
N38 G29 E90 K0 N=E126
N39 E92=E93 E93=E93+E116
N40 G29 E115 N=E99
N41 G22 N=99701 E63=1 E93=E93 E116
N42

```

Instruction for use

1. Refer to figure AN011-1 for the starting point for milling
2. After moving the tool to the first depth (see next point) the circular pocket in the main plane is cleaned out by moving the tool in a circle or a spiral (depending on parameter E72) A number of paths is executed until the final depth is reached.
3. The step per depth is recalculated to an average value, so that each step is executed with the same value. In this way it is avoided that the final step is very small.
4. The points on the spiral are calculated in steps of 6° . To make the operation more (or less) accurate, recalculate E124 in block N21 of macro N99703. Increase (decrease) the value 60 in the denominator to achieve a more (less) accurate operation.
5. With spiral milling the final cleaning step is a circle.
6. Macro N99703 can be used together with the Groundage macros for hole patterns. Refer to Groundage Application Bulletin 02, dated 910220 for a description of some patterns.
7. Refer to chapter Application Notes, section finishing a circular pocket:
 - for a description of the macros N99700 and N99701
 - for instructions how to handle, if an error is found in one of the used macros
8. If the pocket diameter -E64- is negative at the macro call, the call of macro N99700 and the check on E69 are ignored.
9. Refer to the Appendix of this Application Note for the use of macro N99550 (Milling a spiral of Archimedes) as a separate macro.

Example of the use



AN008-3

Figure AN011-2 Roughing a circular pocket

The circular pocket from figure AN011.02 has to be milled and finished. The part program using macro N99703 could be:

```

N2070317
N1  G54
N2  G17
N3  G98 X0 Y0 Z0 1100 J100 K-30
N4  G99 X0 Y0 Z0 1100 J100 K-25
N5  Z10 T8 M67 E60=17 (ENDMILL DIAMETER)
N6  E48=14 E50=50          E51=50 E52=0 E62= 12 E64=60 E69=20
N7  E70=1 E71=1 E72=3 E73=2000 E74=500 E75=1
N8  E76=50 E77=3.5 E78=0 E79=1
N9  G22 N=99703
N10 G0 Z100 M30

```

Explanation

N3/4: Define the window (G98) and material dimensions for the graphical display of the pocket on the control.

N5: Load the endmill and define the plane.

N6: Enter the parameters for the location and the dimensions of the pocket.

N7-N8: Enter the parameters for milling the pocket

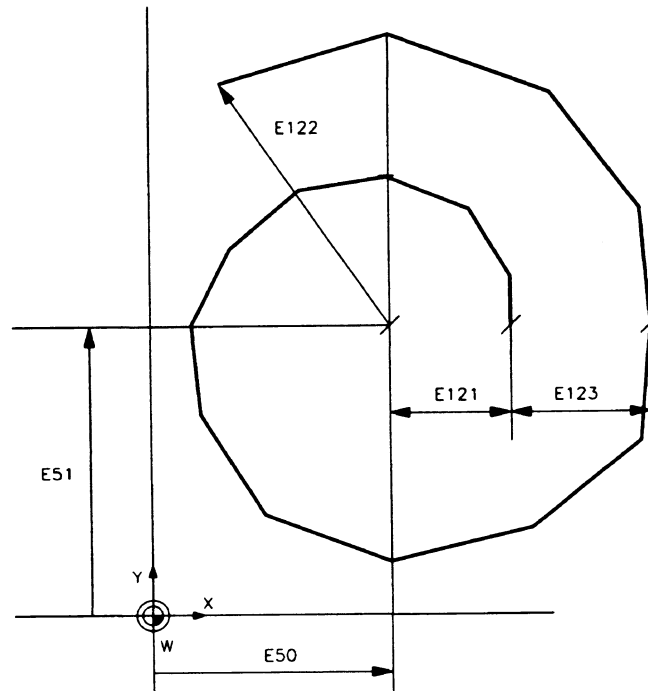
N9: Call the macro for milling the pocket.

N10: Retract the tool; end of the program.

APPENDIX

Macro N99550 (Milling a spiral of Archimedes) can also be used separately. The spiral can be milled either to the inside or to the outside.

Input parameters for the spiral



AN011-3

Figure AN011-3 Meaning of the spiral parameters

E121:	starting radius of the spiral
E122:	ending radius
E123:	radius increase per turn (no sign)
E124:	tolerance value (= deviation on the radius) (no sign)
E125:	= 0 no circle is milled after the spiral <>0 mill a circle after the spiral
E73:	milling feed
E75:	direction of milling looking from the tool towards the workpiece =+1: conventional milling (counter clockwise) =-1: climb milling (clockwise)

Instructions for using macro N99550

1. In the calling macro the tool has:
 - to move to the right starting position and at depth
 - to be retracted after milling the spiral
2. If the starting radius (E121) < the ending radius (E122) the movement on the spiral is to the outside. If E121 > E122, the movement is to the inside.
3. If the difference between starting (E121) and ending radius (E122) is less than the tolerance value (E124), milling the spiral is ignored without an error message.
4. If a circle has to be milled once the spiral is finished, in macro N99550 the tool is moved to the starting point on that circle. The actual circle milling has to be executed in the calling macro.

112.12 A circular pocket with spherical bottom

Introduction

A special macro is designed for milling a circular pocket:

- with a spherical bottom. In this case the milling should be performed with a ball cutter.
- with a spherical wall and a flat bottom. In this case a cylindrical cutter with corner roundings should be used.

Input parameters

E60: Code for the tool axis

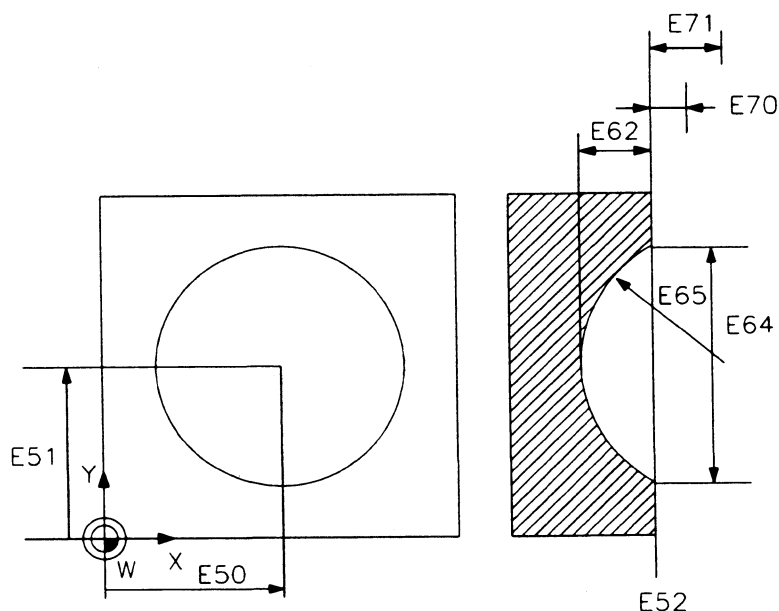
=17 for Z-axis

=18 for Y-axis

=19 for X-axis

Parameter E60 should be set once in the program. As long as the tool axis is not changed or the parameter not used for other purposes, its value remains unaltered and can be used by the macros.

Location and geometry of the pocket



AN012-1

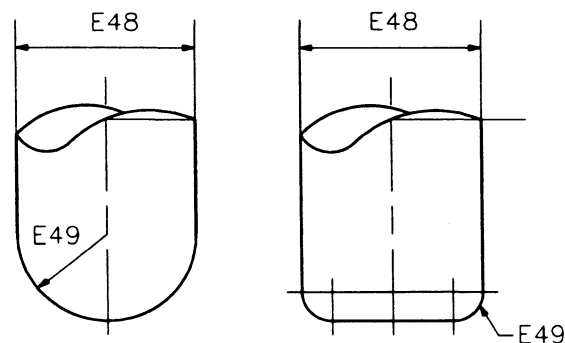
Figure AN012-1 Meaning of the input parameters

- E50: the X-coordinate of the centre of the sphere
part surface position in the toolaxis with G19
- E51: the Y-coordinate of the centre of the sphere
part surface position in the toolaxis with G18
- E52: the Z-coordinate of the centre of the sphere
part surface position in the toolaxis with G17
- E62: depth of the pocket
- E64: diameter of the pocket
- E65: radius of the sphere

Operating parameters

- E70: clearance distance
- E71: the retract distance
- E72: way of roughing the circular pocket
=0 circular milling
=2 spiral milling
- E73: feedrate for the milling operation
- E74: infeed
- E75: direction of milling looking from the tool towards the workpiece
=+1: conventional milling (counter clockwise)
=-1: climb milling (clockwise)
- E76: overlap parameter (in % of tool diameter)
- E77: depth per step (without sign)
- E79: finishing allowance

Tool parameters



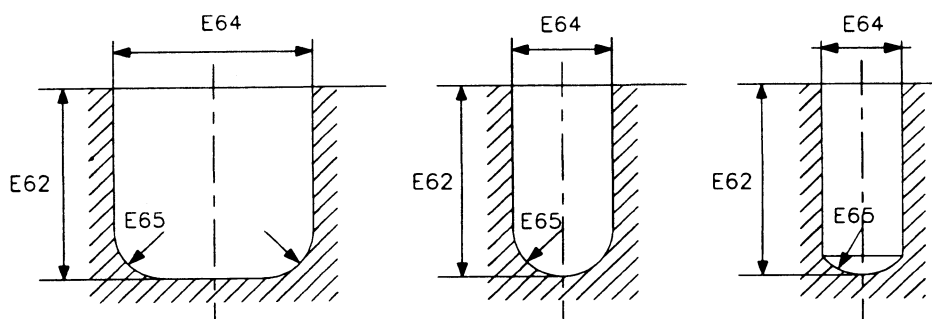
AN012-2

Figure AN012-2 Tool parameters

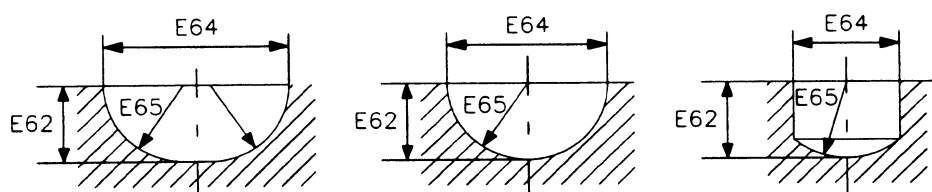
- E48: diameter of the cutter
- E49: = (E48:2) for a ball cutter
= the corner radius for a cylindrical cutter

Remarks:

1. All coordinates and distances are in mm or inches depending if the program is metric or in inches. Coordinates are always absolute.
2. With all distances the sign should be considered.
+: in positive direction of the axis.
-: in negative direction of the axis.
3. The feed is:
with G94 active in mm/min (or inches/min)
with G95 active in mm/rev (or inches/rev).
4. All parameters must be defined at the macro call.
5. The geometry of the pocket is not restricted to the possibility shown in figure AN012.01. The following situations are also possible. They are defined with the same parameters:

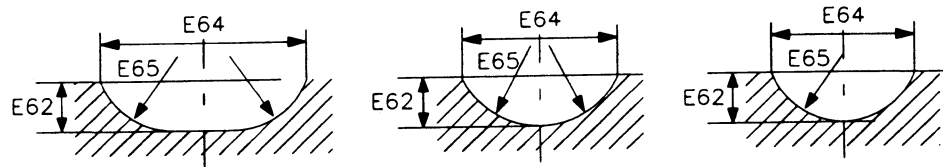
1. E62>E65

AN012-3

Figure AN012-3 Pocket geometry with E62>E65**2. E62=E65**

AN012-4

Figure AN012-4 Pocket geometry with E62=E65

3. E 62<E65

AN012-5

Figure AN012-5 Pocket geometry with E62<E65**Output parameter**

E89: error indicator
 =1: an error detected in the macro
 =0: no error

Local parameters

E90: for the unconditional jump
 E91: used with the relational expression
 E92 - E99: different meanings with the cycles
 E100 - E110: special meanings in the pattern macros
 E111 - E119: special meaning in macro N99701
 E121 - E125: input parameters for the spiral macro N99550
 E126 - E127: special meaning in macro N99703
 E128 - E139: special meaning in macro N99550
 E130 - E133: special meaning in macro N99704
 E140 - E149: special meaning in macro N99704

Error detected

Plane - E60 - not correctly defined

Tool diameter - E48 - not defined

Diameter of the existing hole - E69 - greater than the pocket diameter

Tool should be a ball cutter

Corner radius - E49 -not correctly defined

Calling macros

N99700 (Setting tool axis)
 N99701 (Finishing a circular pocket)
 N99703 (Roughing a circular pocket)

N99550 (Milling a spiral of Archimedes)
 (called via N99703)

Called by

N99580 general pattern macro, if any pattern is used.

The macro**N99704 (CIRCULAR POCKET WITH SPHERICAL BOTTOM)**

```

N1  G22 N=99700 E115=0 E130=2*E65 E147=E77 E148=E64
N2  G29 E89 KO N=47
N3  G29 E91=E49>0 E91 N=6
N4  E89=1 M0 (CORNER RADIUS - E49- NOT CORRECTLY DEFINED)
N5  G29 E89 KO N=47
N6  G29 E91=E65>=E49 E91 N=9
N7  E89=1 M0 (RADIUS OF SPHERE -E65- TOO SMALL FOR TOOL)
N8  G29 E89 KO N=47
N9  E140=2*E49 E141=abs(E62) E142= 1 E143=23 E144=E94
N10 G29 E91=E62<0 E91 N=12
N11 E142=1
N12 E93=E97 E118=E95-(E64:2) E145=E65-E141 E149=0 E132=0
N13 G29 E91=E64>E130 E91 N=22
N14 G29 E91=E64=E130 E91 N=19
N15 E145=sqrt(E65*E65-(E64*E64:4)) E146=E65-E145
N16 E131=E141 E146 E143=34 E149=1 E132=E145 E131
N17 G29 E91=E141<=E146 E91 N=19
N18 E143=26 E131=E131+E49-2*E49*E145:E64
N19 G29 E91=E49=E123 E91 N=E143
N20 E89=1 M0 (TOOL SHOULD BE A BALL CUTTER)
N21 G29 E89 KO N=47
N22 E140=E140+E64-E130
N23 G29 E91=E141<E65 E91 N=34
N24 G29 E91=E141=E65 E91 N=30
N25 E131=E141 E65
N26 E146=int(E131:E77+.99) E116=E142*E131:E146 E141=E148 E143=43
N27 E145=E131+E132 E146=E146 1
N28 G29 E90 KO N=43
N29 G29 E91=E149=1 E91 N=34
N30 G29 E91=E77<E49 E91 N=32
N31 E77=E49
N32 E131=E49 E149=1 E132=0
N33 G29 E90 KO N=26
N34 E143=35 E149=E65-E49 E145=E149*E145:E65
N35 E116=E49-E49"E145:E149 E146=1
N36 G29 E91=E116>.01 E91 N=39
N37 E92=E93+E70 E94=E144 E95=E118 E97=E62 E116=0 E141=E140
N38 G29 E90 KO N=44 E143=47
N39 G29 E91=E116<=E77 E91 N=41
N40 E145=E145 (E116 E77) E116=E77
N41 E134=(E149+E145)*(E149-E145) E116=E142*E116
N42 E141=2*sqrt(E134)+E140 E145=(E145+E49)*E149:E65
N43 E92=E93+E70 E94=E92+E116 E95=E118+(E141:2) E97=E93
N44 G22 N=99703 E64= E141 E115=0
N45 G29 E146 N=E143
N46 G29 E90 KO N=29
N47 E64=E148 E77=E147

```

Instruction for use

1. Refer to Groundage Application Note AN011, date 01-07-1991, for a description of macro N99703 for roughing a circular pocket.
2. Refer to Groundage Application Note AN008, dated 20-06-1991:
 - for a description of the macros N99700 and N99701
 - for instructions how to handle, if an error is found in one of the used macros

3. Nesting of macros

Notice that from macro N99704 the macros are 3 times nested. If macro N99704 is used together with a pattern the nesting is 4 or 5 times. The maximum allowable macro nesting in the Groundage control is 8 times, so be careful that this maximum is not violated.

Example of the use

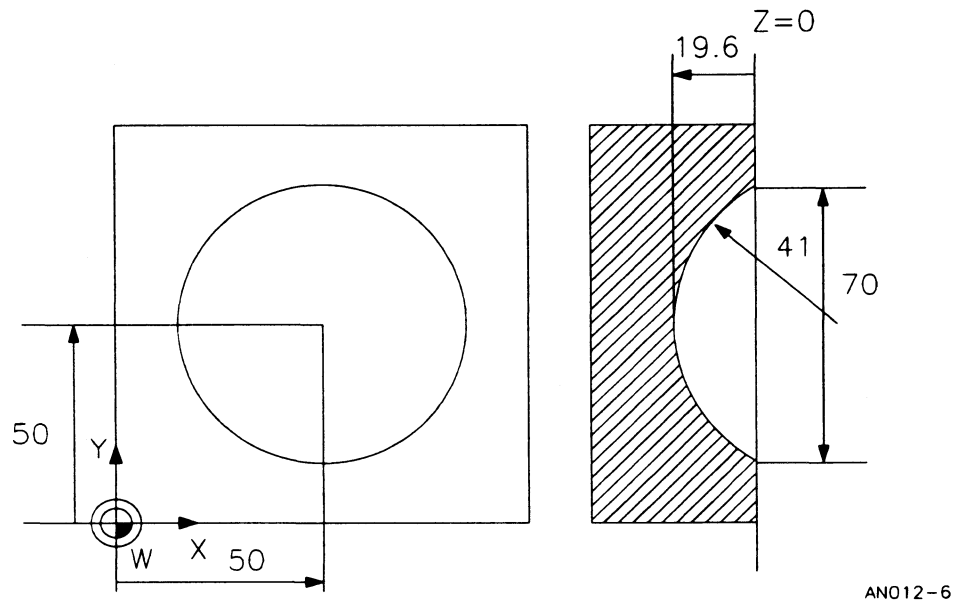


Figure AN012-6 Circular pocket with spherical bottom

The pocket from figure AN012-6 has to be milled. The part program using macro N99704 could be:

```

N20704
N1  G54
N2  G17
N3  G98 X0 Y0 Z0 I100 J100 K-30
N4  G99 X0 Y0 Z0 I100 J100 K-25
N5  Z10 T8 M67 E60=17 (BALLCUTTER DIAMETER 6)
N6  E48=6 E49=3 E50=50 E51=50 E52=0 E62=-19.6 E64=70 E65=41 E69=10
N7  E70=1 E71=1 E72=0 E73=2000 E74=500 E75=1
N8  E76=50 E77=3.5 E78=0 E79=1
N9  G22 N=99704
N10 G0 Z100 M30
  
```

Explanation

N3/4: Define the window (G98) and material dimensions for the graphical display of the pocket on the control.

N5: Load the endmill and define the plane.

N6: Enter the parameters for the tool, the location and the dimensions of the pocket.

N7-N8: Enter the parameters for milling the pocket

N9: Call the macro for milling the pocket.

N10: Retract the tool; end of the program.

113. Index

- 3D-Tool correction G141 131, 141, 313
 A circular pocket with spherical bottom . . 659
 About this manual 15
 Absolute/incremental programming
 G90/G91 293
 Activate cycle G79 249
 Appendix graphical support 310, 363
 Application notes 16, 603, 619, 626, 631, 637,
 640, 645, 648, 651, 657
 Automatic tool change M6 467
 Auxiliary function H 457
 Axes configurations on machine tools . . . 6, 10
 Axis parallel measuring movement G45 . . 145
 Back boring cycle 611, 614
 Begin contour description G199 375
 Begin inner/outer contour description
 G197/G198 367
 Block delete 388
 Block search 323, 389
 Bolt hole circle G77 239
 Boring cycle G86 271
 Boring cycle without dragline 607
 Cancel tool radius compensation G40 . . . 127
 Cancel/activate G52 zero point shift
 G51/G52 175
 Cancel/activate geometric calculations
 G63/G64 199
 Cancel/activate scaling or mirror imaging
 G72/G73 227
 Cancel/activate zero point shift G53/G54 -
 G59 179
 Cartesian coordinates 12, 21, 24, 28, 40-42, 5
 7, 64, 213-215, 245, 297, 352, 357
 , 376, 413, 419
 Change tool compensation values M67 . . 491
 Checking on tolerances G49 161
 Circular interpolation (CW/CCW) G2/G3 . . 35
 Circular pattern 620, 624
 Circular pocket contour 402
 Circular pocket milling cycle G89 287
 Combining a linear coordinate and angle . . 14
 Companion manuals 1
 Conditional jump G29 119
 Contents of each section 15
 Deep hole drilling cycle G83 257
 Defining coordinates 12
 Definition of workpiece blank as a box
 G99 311
 Drilling cycle G81 253
 Dwell time G4 53
 Editing 6, 388
 Enable/disable feed override G25/G26 . . 107
 End contour description G196 365
 End of partprogram M30 485
 Error messages 147, 156, 390, 439, 448
 Examples of pocket descriptions 405
 Feed-in point 400
 Finishing a circular pocket or island 635
 Functions. 9, 17
 G200 . 77, 383-389, 392, 394, 397, 399, 407, 4
 09, 410, 413, 414, 415, 417-419, 4
 22, 425, 426, 429, 431, 436, 588,
 601
 G201 . 123, 383-389, 392, 394, 397, 399, 400,
 403, 404, 405-407, 409, 410, 415,
 417, 418, 420, 422, 426, 430, 436,
 453
 G202 . 384-389, 391, 394, 399, 400, 404-407,
 409, 411, 415, 417, 418, 423, 427,
 430, 437, 446, 601
 G203 . 383, 385-389, 392, 394, 397-399, 404-4
 06, 409, 410, 413-415, 417-420, 4
 22, 426, 431-436
 G204 . 385-387, 389, 392, 394, 404-407, 409,
 410, 414, 415, 417, 418, 422, 426,
 432-436
 General format of the macro for a circular
 feed-in 401
 General programming information 3
 General remarks 421
 Generation 388
 Geometric calculations with continuous
 movements 16, 511
 Geometric calculations with
 non-continuous movements
 G64 553
 Graphic window definition G195 363
 Graphic window definition G98 309
 Grid pattern 625, 628, 630, 633
 Groove milling cycle G88 281
 Inch/metric programming G70/G71 225
 Incomplete programming 389
 Incremental/absolute zero point shift
 G92/G93 299
 Index . . . 4, 183, 245, 343, 344, 347, 348, 667
 Intervention 323, 389
 Introduction . 1, 447, 603, 607, 611, 615, 620,
 625, 630, 635, 643, 647, 652, 660
 Island contour 386, 421, 430
 Linear feed-in movements 403
 Linear interpolation G1 23
 Linear measuring movement G145 321
 Linear pattern 615
 Machining macro 400, 403
 Macro call G22 99, 389
 Macro for finishing a pocket contour 400
 Mainplane XY, tool Z G17 87
 Mainplane YZ, tool X G19 95
 Manual tool change M66 489
 Measure tool dimensions G45+ M25 151
 Measuring a full circle G46 153
 Milling a circular groove 643
 Mirror image of a pocket 386

- Operating section 388
- Operation mode change 389
- Optional program stop M1 461
- Oriented spindle stop M19 477
- Part programming 3
- Partprogram structure 385
- Philosophy and purpose of the manual 15
- Pocket contour 386, 387, 400, 402
- Pocket contour description 385, 413, 417, 425
- Pocket, starting points and finishing
 macros 388
- Point definition G78 245
- Polar coordinates 13, 20, 24, 30, 41, 47, 63, 6
 5, 85, 88, 92, 96, 213, 214, 215, 2
 45, 248, 294, 297, 302, 368, 370,
 376, 413, 419, 507, 511
- Probe calibration G46+ M26 159
- Processing measuring results G50 167
- Program blocks 5, 63, 121, 443, 468
- Program call G23 105
- Program stop M0 459, 640
- Program words 4, 15, 502
- Programmable absolute position G74 235
- Rapid traverse G0 19
- Read probe status G148 341
- Read tool data and zero offset G149 343
- Reaming cycle G85 267, 603
- Reaming cycle with separate outfeed 603
- Remarks 28, 44, 421, 604, 608, 612, 637, 64
 4, 654, 662
- Removal 279, 290, 388, 397
- Repeat function G14 83
- Rhomboidal pattern 647, 650
- Rotation of an island around the starting
 point 420
- Roughing a circular pocket 652, 656, 658
- Same pocket in another program 387
- Select feedrate unit G94/G95 307
- Select negative/positive tool direction
 G66/G67 221
- Select spindle speed range
 M41/M42/M43/M44 487
- Sequence of the macros on the machine
 403
- Sign of life 388
- Spindle rotating clockwise/counter
 clockwise M3/M4 463
- Spindle speed S 493
- Spindle stop M5 465
- Spline interpolation G6 55
- Staggered grid pattern 630, 633
- Starting points macro 399, 403
- Switch off coolant supply M9 473
- Switch on / off a measuring probe
 M27/M28 483
- Switch on coolant supplynr. 2 / Nr. 1.
 M7/M8 469
- Switch on nr. 1 coolant and rotate spindle
 CW/CCW M13/M14 475
- Tangential exit G62 193
- Tapping cycle G84 263
- Teach-in 364, 365, 370, 378, 388
- The controller 1, 16, 170, 175, 177, 188, 352,
 441
- The programmer 5, 6, 9, 77, 190, 195, 221, 23
 0, 231, 250, 265, 383, 384, 386, 3
 99-402, 414, 461, 496, 546, 547, 5
 49
- Tool number T 495
- Tool radius comp. To/past endpoint
 G43/G44 141
- Tool radius compensation (left/right)
 G41/G42 131
- Translation, rotation and mirror image of a
 pocket 386
- Usage of the generated macros 399
- Write tool data and zero offset G150 347
- Zero points 7, 9, 177, 180, 183, 184, 237